Spikemark™ v3.0

User Manual

Spikemark™, Stagehand™ and Showstopper™ are all trademarks of Creative Conners, Inc. All rights reserved.

This page has intentionally been left blank.

Table of Contents

Welcome to Spikemark 7
A Word about Safety 7
System Requirements 8
Installing Spikemark (in Windows 8) 8
Online Tutorials 14
Spikemark Highlights 14
Simulator - *New in Spikemark 3* 14
Machine Library - *New in Spikemark 3* 15
WATCHOUT Integration - *New in Spikemark 3* 15
Spikes 15
Time-Based Cues 16
Streamlined Interface 16
3-Dimensional Stage View 18
Quick Start in the Simulator 19
Writing Your First Cue 19
Adding a Second Machine to the Show 22
Writing Cues with Multiple Machines 26
Using Spikes 33
Moving Real Machinery with Spikemark 40
Setting Up Physical Connections 40
Setting Up a Network Connection (in Windows 8) 41
Setting the IP Address of Your Computer 41
Adding Machinery to the Show 45
Add a Pre-Configured Machine from the Machine Library 45
Add a New Blank Machine 46
Removing a Machine from the Show 47
Establishing a Network Connection to your Machine 49
Confirming Encoder Feedback 51
Setting Soft Limits 53
Writing Two Sample Cues 54
Running the Two Sample Cues 56
Basic Tuning 59
Conclusion 60
Spikemark's Main Screen 61
Stage Model Viewer 61
Stagehand Editing Pane 61
Cue Controller 61
Cue Grid 61
Splitter Panes 62
Simulator and Live Mode – *New in Spikemark 3!* 63
The Stagehand Motor 65
Adding a New Stagehand Motor 65

Stagehand Motor Configuration 65
Renaming a Motor 67
Motor Status 68
Manually Moving a Motor Using the On-Screen Controls 69
Editing Motor Direction Labels – *New in Spikemark 3!* 70
Scaling Motor Position 71
Setting Position Scale for a Winch 71
Setting Position Scale for a Turntable 74
Setting Motor Software Limits: The Min Reverse and Max Forward Positions 76
Resetting the Motor's Position Value 78
Motor Tuning 79
Proportional Gain 80
Derivative Gain 80
Derivative Sampling 81
Integral Gain 81
Integral Limit 82
Filter Loaded 82
Setting the Max Position Error 85
Setting the Motor Cue Completion Mode 86
Target Tolerance 86
Strict Timing 86
Setting a Motor's Active Mode 88
Changing the Status Refresh Rate 90
The Machine Library – *New in Spikemark 3!* 93
Adding a Pushstick Winch to the Show 93
Adding a Curtain Call Winch to the Show 95
Adding a Revolver to the Show 97
Adding a Custom Machine to the Machine Library 99
Deleting a Machine from the Machine Library 101
The Stagehand FX 103
Adding a New Stagehand FX 103
Stagehand FX Configuration 105
Renaming an FX 105
Renaming FX Inputs and Outputs 106
FX Status 108
Manually Activating FX Outputs 109
Setting the FX Active Mode 109
FX Input Action 112
Deactivating Stagehand FX Action 112
Soft Stopping FX Action 113
Run Next Cue FX Action 114
Removing a Stagehand FX Input Action 115
Network Connections 116
Editing Stagehand Network Settings 116
Connecting to a Stagehand Motor or FX 118
Disconnecting from a Stagehand Motor or FX 119

Writing Cues 121
Adding Cues 122
Adding Motor Movements to a Cue 123
Motor Movement Parameters 124
Position 124
Speed 125
Acceleration 125
Ramp Time 125
Total Time 125
Changing Motor Position In a Cue 126
Changing How Fast a Motor Moves Within a Cue 127
Specifying the Speed of a Motor Directly 127
Specifying *How Long* the Motor Should Take to Reach the Target Position 128
Changing Cue Time 129
Adding an FX Movement to a Cue 130
Removing a Motor's Movement from a Cue 131
Deleting a Cue 132
Cue Links 134
Parent Cues 134
Child Cues 134
Linking Cues by Completion 135
Linking Cues by Time 137
Linking Cues by Motor Position 138
Linking Cues by FX Input 139
Deleting a Cue Link 140
Multi-Speed/Multi-Position Links 141
Changing the Speed of a Motor—Manually (Without a Link) 141
Changing the Speed of a Motor—Automatically 142
"Ping-Pong" a Motor—Changing a Motor's Position 143
The Stage Model Viewer 145
Navigating the Stage Model Viewer 146
Setting the Stage Dimensions 149
Making a Winch Model 150
Making a Lift Model 153
Making a Curtain Model 156
Making a Turntable Model 159
Spikes 163
Adding Spikes within Cues—Using the Spikes Position Button 163
Adding Spikes within Cues—Using the Stagehand Editing Pane 167
Changing a Spike Position 171
Running Shows 174
Creating a New Show 174
Setting up the Workspace 174
Rearranging the Cue Grid 176
Loading a Cue 177
Unloading a Cue 179

Visual Indicators of Primed and Ready Movements 179
Running a Cue 181
What Happens While a Cue Is Running 182
What Happens as Motor Movements and Cue(s) Complete 184
Deciphering Stagehand Motor Status During a Show 187
Restoring a Cue 189
Dataton WATCHOUT Integration – *New in Spikemark 3!* 193
Troubleshooting 207
Contact Us 207
The Stagehand Network Connection Doesn't Work 207
A Motor Didn't Start To Move When I Ran A Cue 208
Spikemark Showed A Strange Error Message Then I Had To Quit 208
I Can't Find The Log File To Email It To Support 209
A Motor Starts To Run In A Cue But Stops Abruptly Before Getting To Spike 209
The Motor Runs Full Speed In The Wrong Direction 210
Spikemark Not Responding/ Running Slowly 210
Appendix A - Basic Motion Control Concepts 211
Power 211
Position Feedback 211
Motor Tuning 212
Over-Travel Limits 212
Index 213

Welcome to Spikemark

Welcome to Spikemark, the software designed for theatrical automation. Spikemark is built to make moving scenery easy and intuitive. As today's shows grow increasingly complex, Spikemark offers a simple solution to bring the excitement of motorized scenery to every production. Using a Windows® PC and standard Ethernet hardware, you can have a motion control system that is easy to set-up and expand while being incredibly economical. Whether you need to move a deck winch, turntable, roll drop, or any other effect, Spikemark allows you to focus on the custom aspects of your production by providing a standard way to control motion.

Using **Simulator**, new in Spikemark 3.0, you can write and run cues on a virtual stage without being physically connected to any machinery. Now you can visualize automated effects in your office, in a design meeting, or on your couch, letting you easily experiment with moving scenery.

A Word about Safety

The spectacle created by motorized scenery is fantastic. Motion control allows huge scene changes to operate smoothly in a matter of seconds. But this power deserves the utmost caution.

Unlike operating a computerized lighting console or sound system, an automation system can pose a serious danger to performers and technicians. Large pieces of scenery moving at high speed can easily injure, even fatally, any person that is unwittingly caught in its path. Therefore, a large responsibility is placed on you, as the automation operator, to be alert and aware of the people that may be in harm's way.

NOTE: The GO button demands a high level of respect.

Spikemark assists the operator in maintaining safety by providing constant information of every motor's position and status. A hard-wired emergency stop system—Showstopper—provides an immediate way for the operator to shut down all the motion on the stage. Ultimately, it is the operator's judgment and ability to see any potential hazards that will keep the show safe.

System Requirements

Spikemark can be used in two modes: **Simulator** or **Live**. **Simulator** will enable you to set up a show, program cues, and run cues on a 3D virtual stage. In Live mode, you can connect Spikemark to real machinery and run cues that make actual scenery move.

The free version of Spikemark can use all of the features of **Simulator**. To run a show in **Live** mode you will need to purchase a license key. If you have a license key, you can flip between **Simulator** and **Live** mode easily.

To effectively use Spikemark in **Simulator** mode you will need:

- A Windows® compatible PC (1.4 GHz or faster)
- A dedicated graphics processor (recommended)
- Windows® Vista or better (Windows® 8 recommended)
- Microsoft's .Net 4.0 framework
- 10MB of free storage space

To run a show with Spikemark in **Live** mode you will additionally need:

- A network interface card (or built-in networking)
- An Ethernet hub
- Ethernet cables (1 for every device on the network)
- One or more Stagehand motor controllers with electric motors (winches, turntables, etc.)
- A Showstopper emergency stop base station

Installing Spikemark (in Windows 8)

1. Download the most recent installer from the Creative Conners site: http://creativeconners.com/the-brains/download

2. Double-click the **SpikemarkSetup** file (Figure 1).

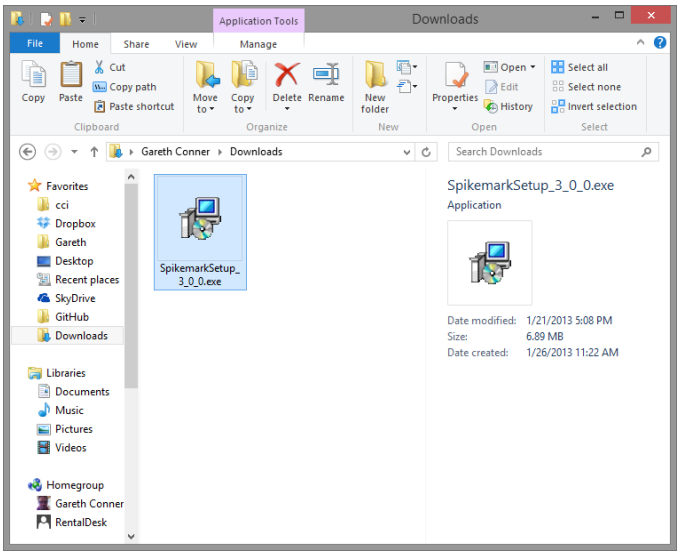


Figure 1

3. In the **User Account Control** window, click **Yes**.

4. In the **Welcome to Spikemark Setup Wizard** window, click **Next** (Figure 2).

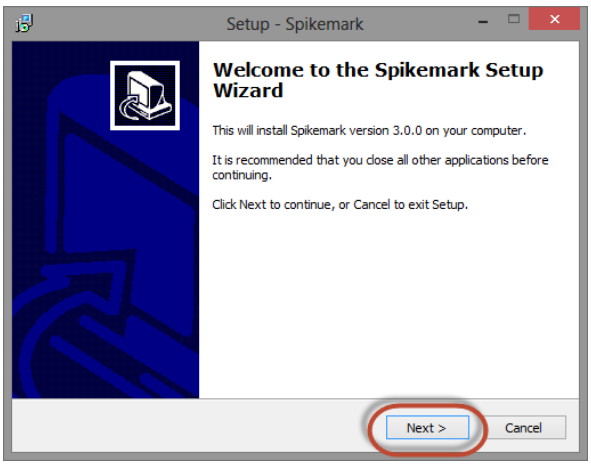

Setup - Spikemark
Welcome to the Spikemark Setup Wizard
This will install Spikemark version 3.0.0 on your computer.
It is recommended that you close all other applications before continuing.
Click Next to continue, or Cancel to exit Setup.
Next >
Cancel

Figure 2

5. In the **License Agreement** window, read the agreement, select **I Agree** and click **Next** (Figure 3).

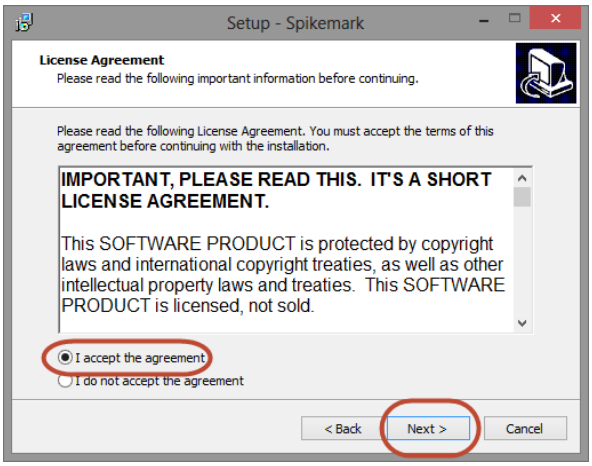

Setup - Spikemark
License Agreement
Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
IMPORTANT, PLEASE READ THIS. IT'S A SHORT LICENSE AGREEMENT.
This SOFTWARE PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. This SOFTWARE PRODUCT is licensed, not sold.
I accept the agreement
I do not accept the agreement
< Back
Next >
Cancel

Figure 3

6. The **Select Destination Location** window will launch. You can either accept the default location (recommended) or choose a custom spot on your computer to install Spikemark. Click **Next** (Figure 4).

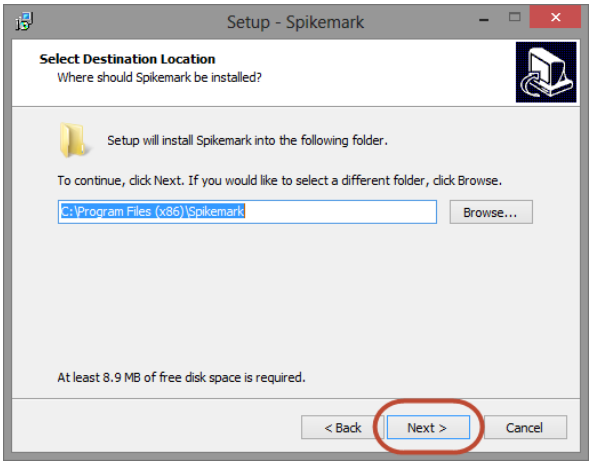

Setup - Spikemark
Select Destination Location
Where should Spikemark be installed?
Setup will install Spikemark into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files (x86)\Spikemark
Browse...
At least 8.9 MB of free disk space is required.
< Back
Next >
Cancel

Figure 4

7. The **Select Start Menu Folder** window launches. Click **Next**. (Figure 5)

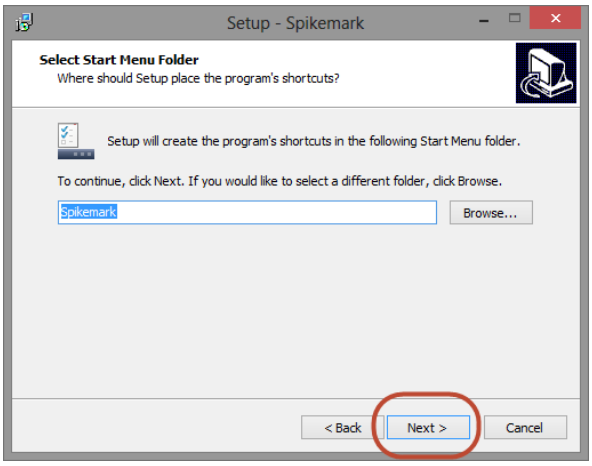

Setup - Spikemark
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
Spikemark
Browse...
< Back
Next >
Cancel

Figure 5

8. If you would like to have an icon on your desktop that you can use to launch Spikemark quickly, check **Create a desktop icon**. Click **Next**.(Figure 6)

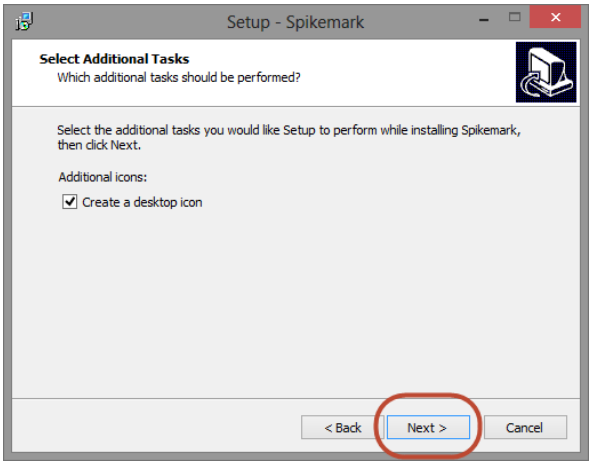

Setup - Spikemark
Select Additional Tasks
Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing Spikemark, then click Next.
Additional icons:
Create a desktop icon
< Back
Next >
Cancel

Figure 6

9. With all the options configured, Spikemark is ready to be installed. Click **Install**.(Figure 7)

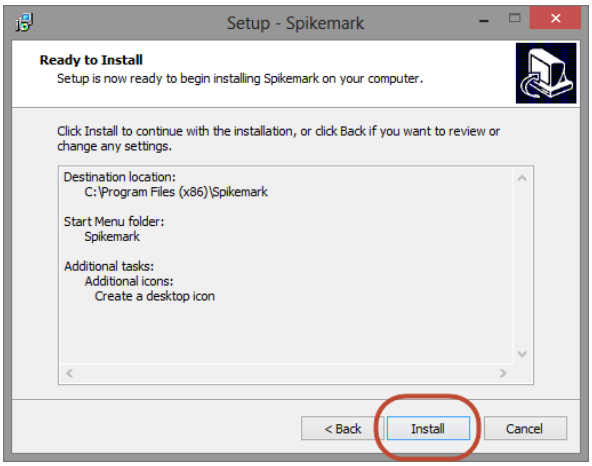

Setup - Spikemark
Ready to Install
Setup is now ready to begin installing Spikemark on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
C:\Program Files (x86)\Spikemark
Start Menu folder:
Spikemark
Additional tasks:
Additional icons:
Create a desktop icon
< Back
Install
Cancel

Figure 7

10. The **Installing Spikemark** window will launch (Figure 8).

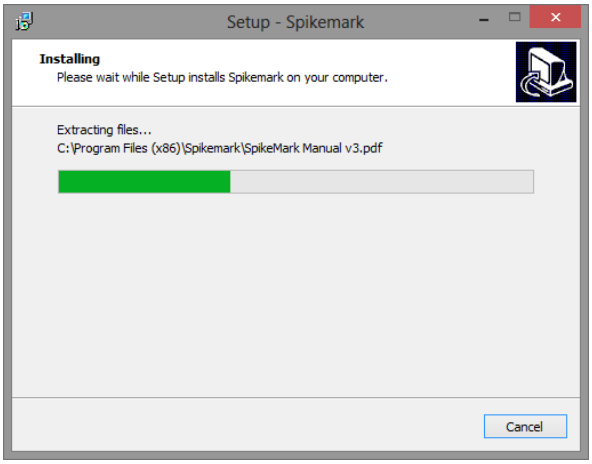


Figure 8

11. In the **Installation Complete** window check the **Launch Spikemark** option if you want to start using Spikemark immediately. Uncheck **Launch Spikemark** if you want to run it later. Click **Finish** (Figure 9).

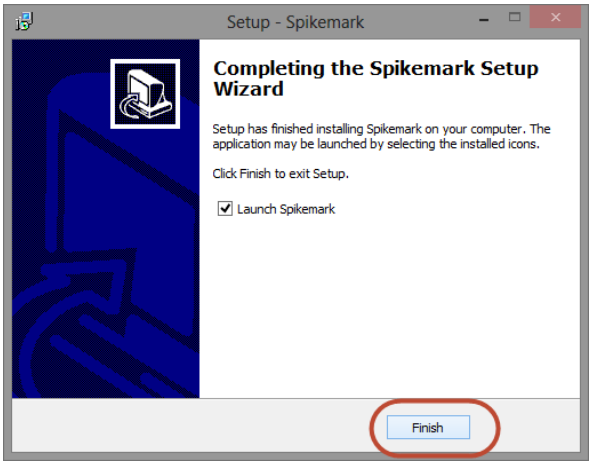

Setup - Spikemark
Completing the Spikemark Setup Wizard
Setup has finished installing Spikemark on your computer. The application may be launched by selecting the installed icons.
Click Finish to exit Setup.
Launch Spikemark
Finish

Figure 9

NOTE: If Microsoft's .Net 4.0 framework is not installed on your system, the installer will direct you to a Microsoft update page where the framework can be downloaded. Also, check Windows Update regularly to keep your Windows operating system up to date.

12. After the Spikemark installation is complete, launch the Spikemark program. The license registration dialog box will appear. If you downloaded the free version of Spikemark and only want to experiment with Spikemark in **Simlator** mode, click the **Simulator-Only Mode** button. If you want to run Spikemark in **Live** mode, enter your **User** and **Password** in the **Registration** window (Figure 10).
*The **User** and **Password** are provided on the inside cover of your Spikemark installation USB key tin.*

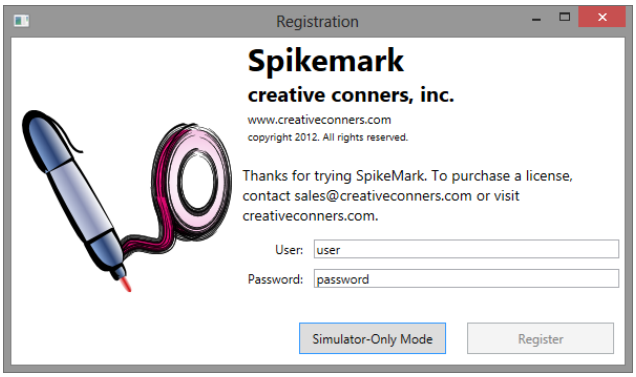


Figure 10

13. If you entered a **User** and **Password**, then click **OK** in the **Thank you** dialog box (Figure 11) to get started with the Spikemark program.

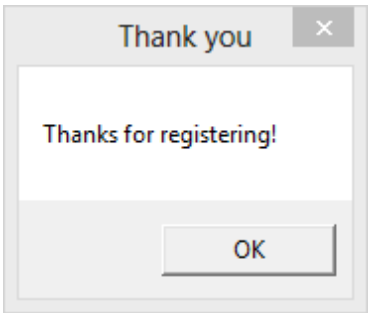


Figure 11

Online Tutorials

There is a selection of video tutorials on the Creative Conners website that cover all the basic uses of Spikemark. http://creativeconners.com/resources/videos/training-videos

Spikemark Highlights

Spikemark has a number of great features, improving on Creative Conner's legacy of providing the easiest software to use for stage automation. Spikemark's name also highlights one its handiest features: Spikes. The following are a selection of old favorite features and some of the great new improvements found in Spikemark 3.

Simulator - *New in Spikemark 3*

Simulator, as the name suggests, simulates physical machinery on a virtual stage letting you run cues and jog scenery around without being connected to any physical motors. **Simulator** is a great way to get familiar with Spikemark without the need to set up heavy equipment. **Simulator** is also an excellent pre-visualization tool, you can use it to work through different cueing ideas at the tech table, in a design meeting, or on your couch (Figure 12).

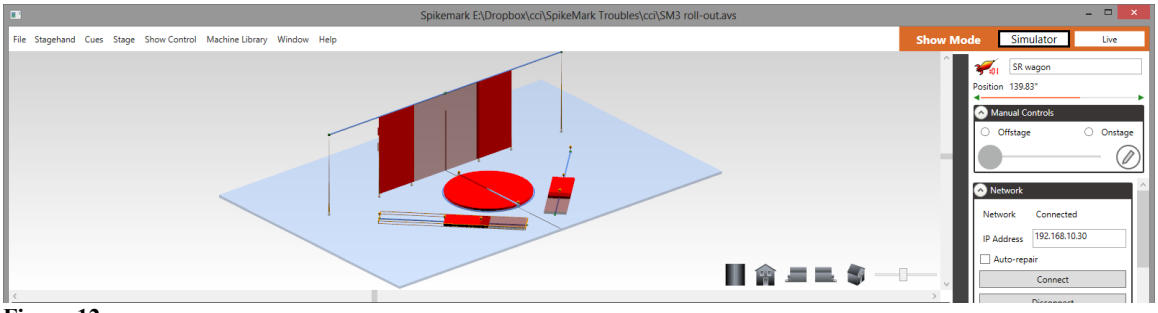


Figure 12

Machine Library - *New in Spikemark 3*

Spikemark's Machine Library makes it easy to start a new show by dragging and dropping pre-configured machines in the main window. By default, the Machine Library is loaded with Pushstick, Revolver, and Curtain Call machines, but you can easily add your own machines to the library. Using the Machine Library eliminates tedious parameter configuration and gets you writing cues fast (Figure 13).

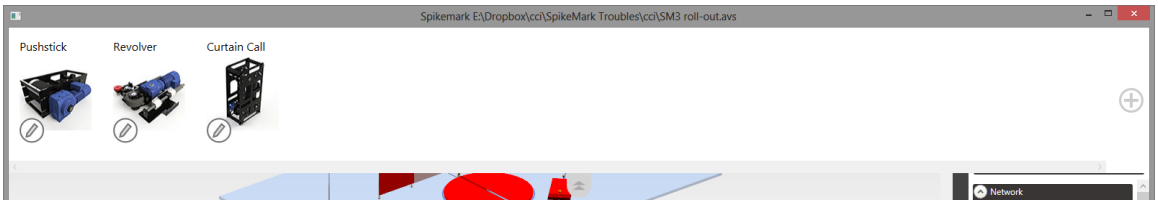


Figure 13

WATCHOUT Integration - *New in Spikemark 3*

Projection is a big part of many shows today. The new WATCHOUT integration feature makes it easy to synchronize projected images with moving scenery by streaming motor position information over the network to Dataton WATCHOUT systems (Figure 14).

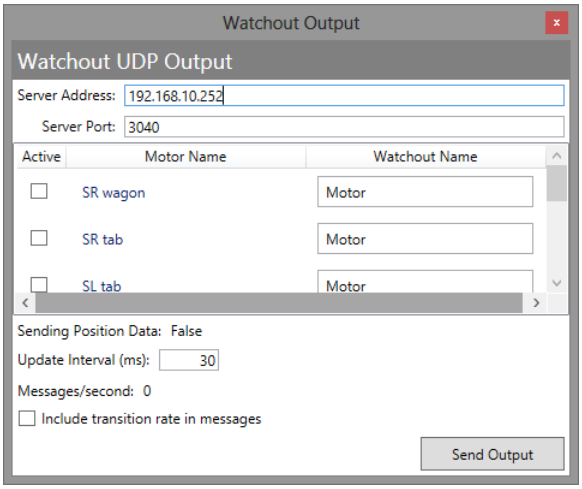


Figure 14

Spikes

The **Spikes** feature is designed to mimic the usefulness of real spike tape. Rather than trying to remember bizarre cue position numbers (such as 251.13"), you can now assign names to commonly used positions ('on stage' = 251.13"). When writing cues, you can select a position from the motor's defined spikes (Figure 15).

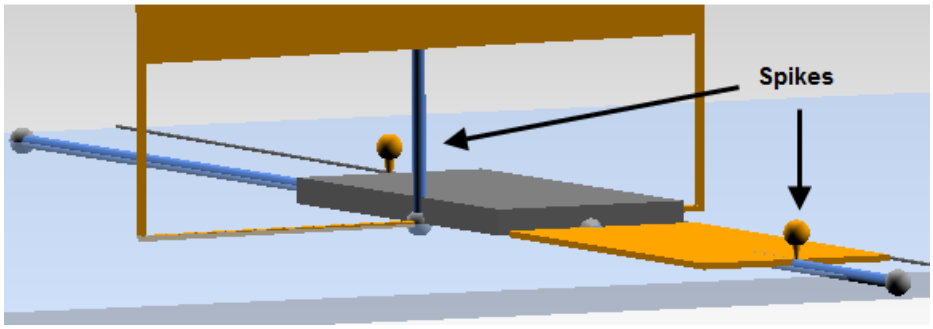


Figure 15

Often during the rehearsal process, the spike positions need to be tweaked. If you were physically pushing the wagons around, you would just move the spike tape on the stage floor to a new position. Similarly, with the new **Spikes** feature, you simply edit the position of the spike in one place and all cues that use the spike are automatically updated, saving you from manually tracking changes through the show.

Time-Based Cues

Cues can be written using *time* parameters or speed and acceleration values. You can write cues by specifying how long the cue should run, and how long it should take for the motors to ramp up to speed (Figure 16). Quickly writing cues that match the timing of the show helps reduce tech-table stress.

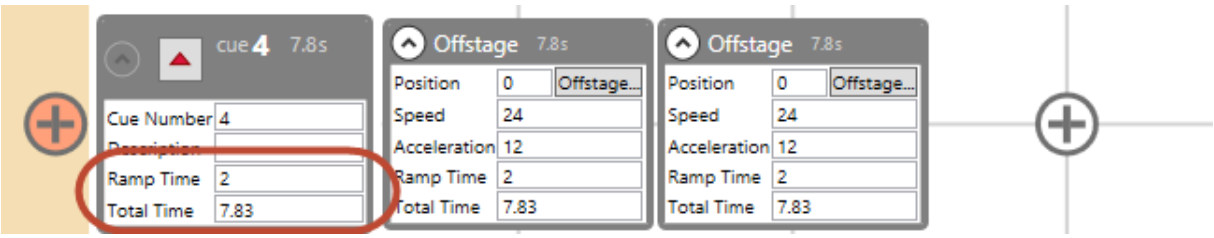


Figure 16

Streamlined Interface

Spikemark's interface is simple and easy to navigate (Figure 17). The main window has "Splitter Panes" (Figure 18), which allow you to create a computer workspace that works best for you. More commonly used features have been rearranged to make them easier to reach.

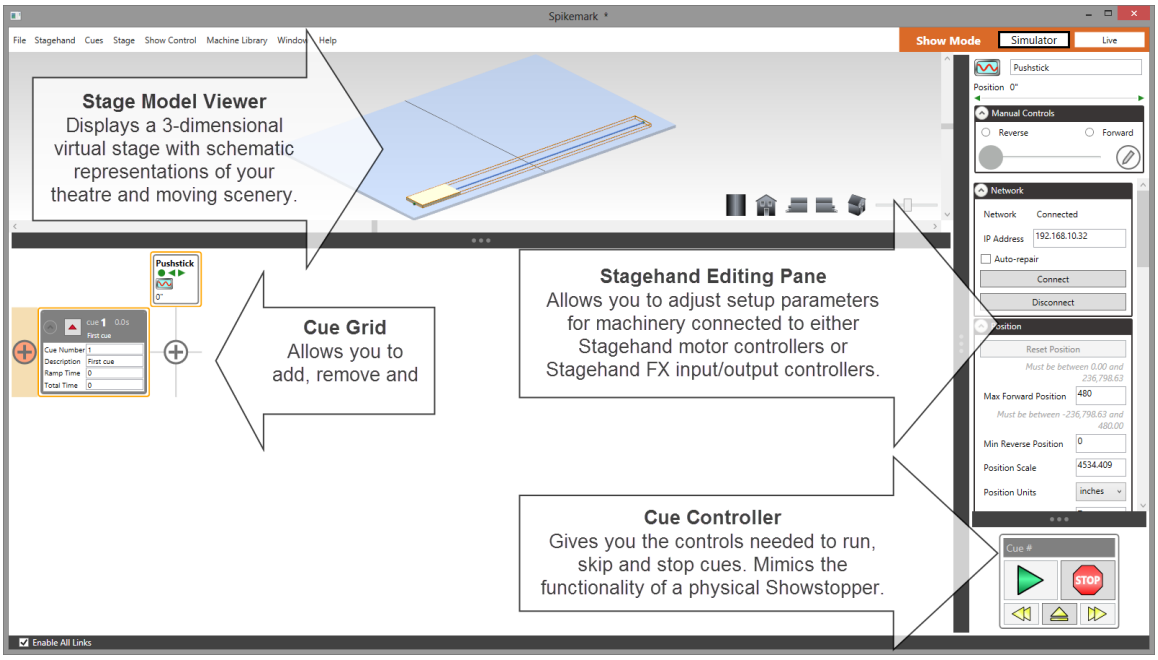

Stage Model Viewer
Displays a 3-dimensional virtual stage with schematic representations of your theatre and moving scenery.
Cue Grid
Allows you to add, remove and
Stagehand Editing Pane
Allows you to adjust setup parameters for machinery connected to either Stagehand motor controllers or Stagehand FX input/output controllers.
Cue Controller
Gives you the controls needed to run, skip and stop cues. Mimics the functionality of a physical Showstopper.

Figure 17

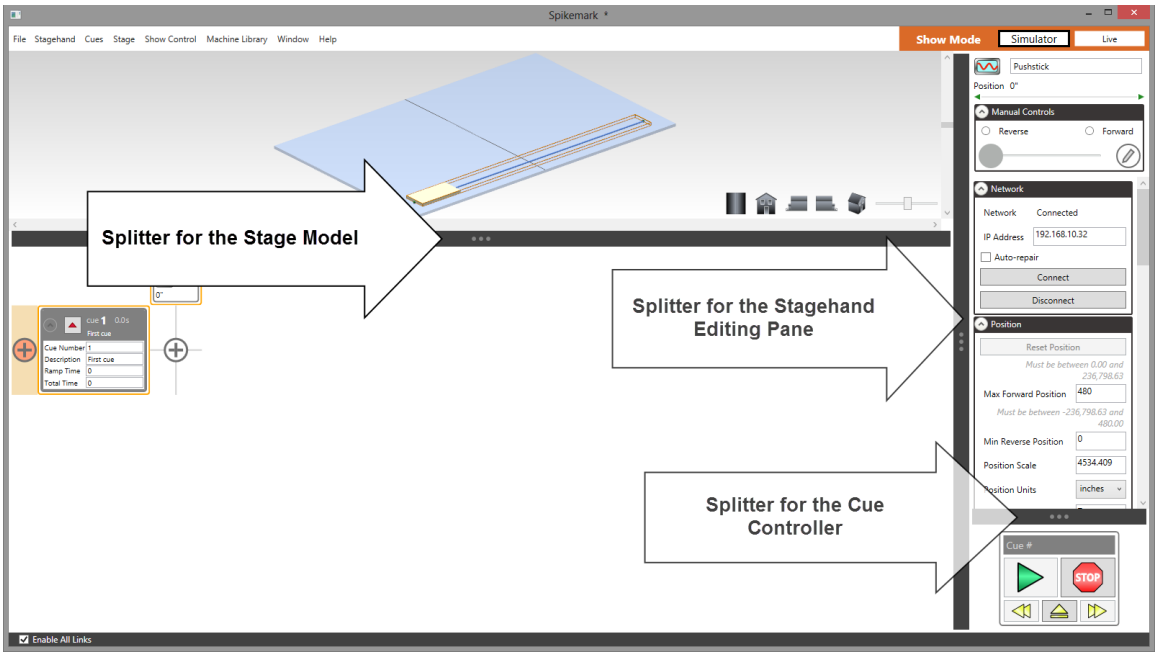


Figure 18

3-Dimensional Stage View

The Spikemark **Stage Model Viewer** gives a three-dimensional view of the theatre space (Figure 19). Wagons and turntables can be rotated along any axis, making it easy to view lifts, curtains, and deck winches in a more realistic fashion.

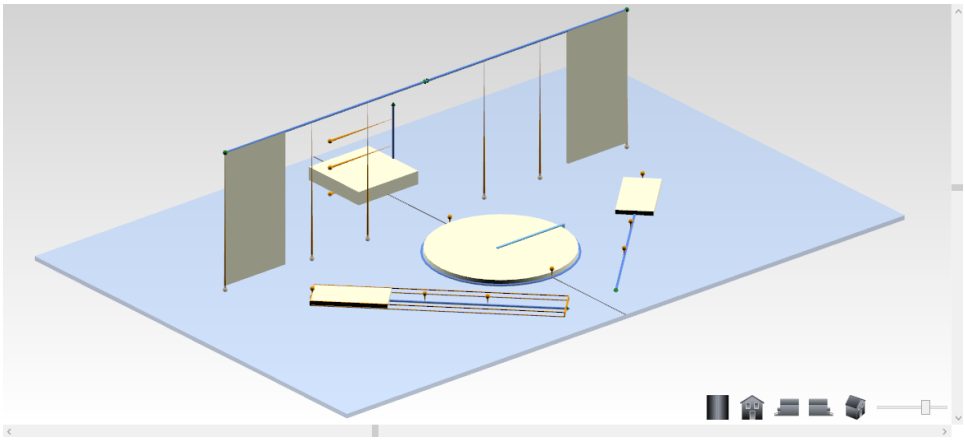


Figure 19

Quick Start in the Simulator

Spikemark has always been easy to use, but with the new **Simulator** it is even easier to get started. Whether you just want to play around in the **Simulator**, or you are itching to hook up some real motors, the **Simulator** is the best way to get familiar with Spikemark. Let's dive right in and write some cues!

Writing Your First Cue

1. Launch Spikemark either from the **Desktop** (Figure 20) or the **Windows 8 Start Screen** (Figure 21).

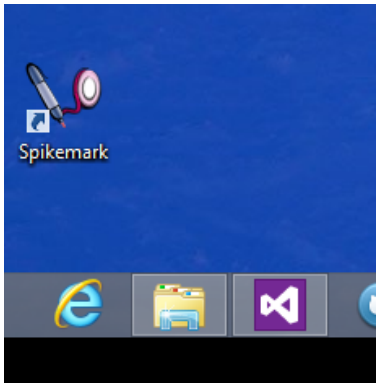


Figure 20

Figure 21

2. Confirm that the **Simulator** mode button is pressed (Figure 22).

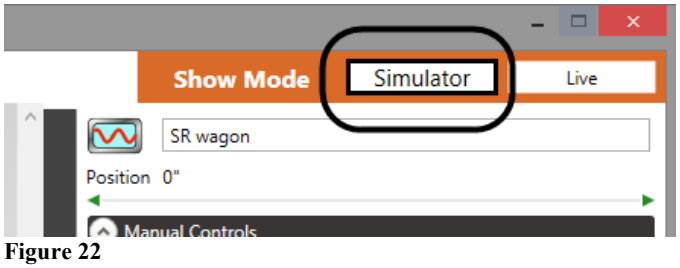


Figure 22

3. When Spikemark first opens the show will contain one wagon named *Pushstick* that is placed downstage by default, and a single cue (Figure 23). We can change the name and position of this wagon (or remove it completely), but for now let's just leave it there.

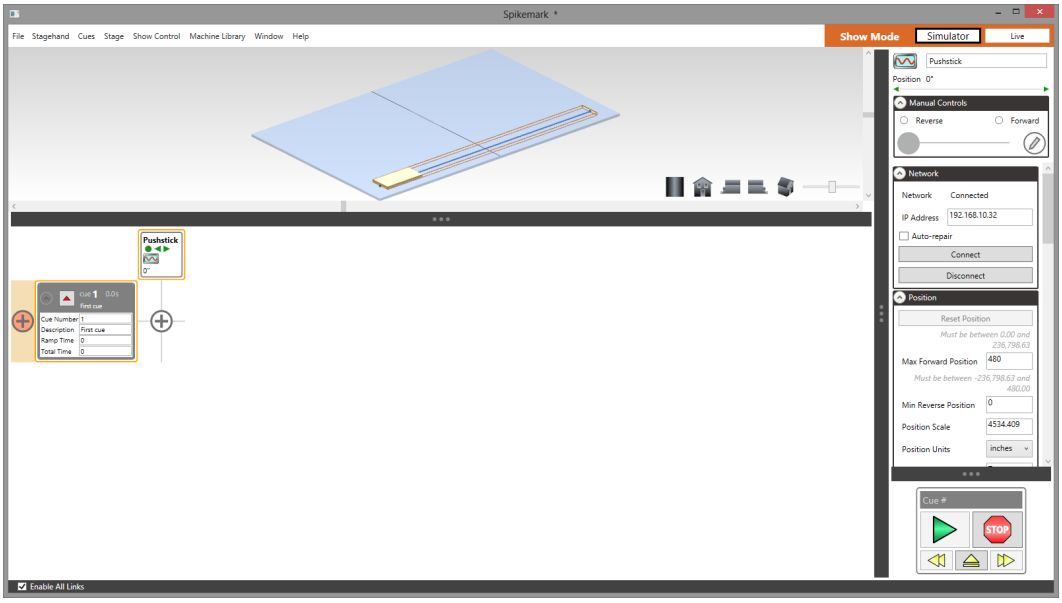


Figure 23

4. To add a movement in *Cue 1*, click on the plus button at the intersection of *Pushstick* and *Cue 1* (Figure 24).

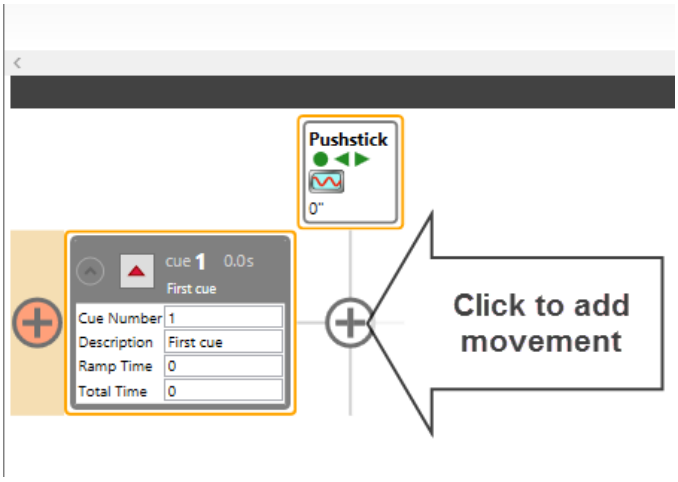


Figure 24

5. A **Movement** panel will appear with parameters for adjusting the speed and position of the motor that is moving the wagon. Enter "240" in the **Position** field to adjust the position of the motor in *Cue 1* (Figure 25).

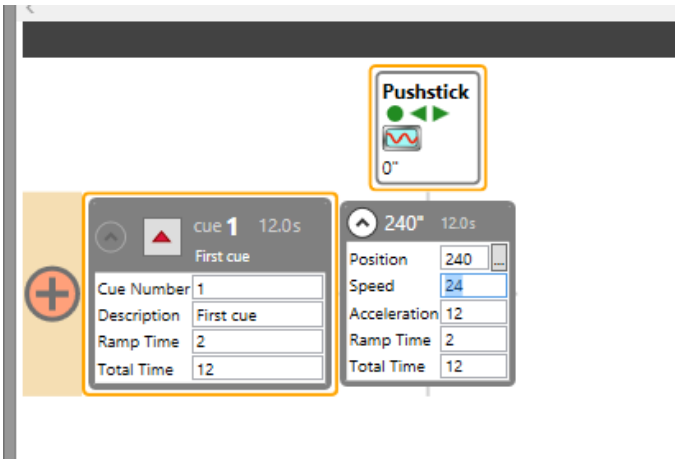


Figure 25

6. Press the **Load this cue** button in the *Cue 1* panel. There are several immediate changes in the interface to indicate that a cue is loaded (Figure 26):

    a. *Cue 1* and its movements are decorated with a red border.

    b. The *Pushstick* panel shows a **rocket icon** with a countdown timer, indicating that it is ready to launch.

    c. The *Pushstick* wagon in the **Stage Model Viewer** turns red and displays a semi-transparent area where the wagon will end the cue.

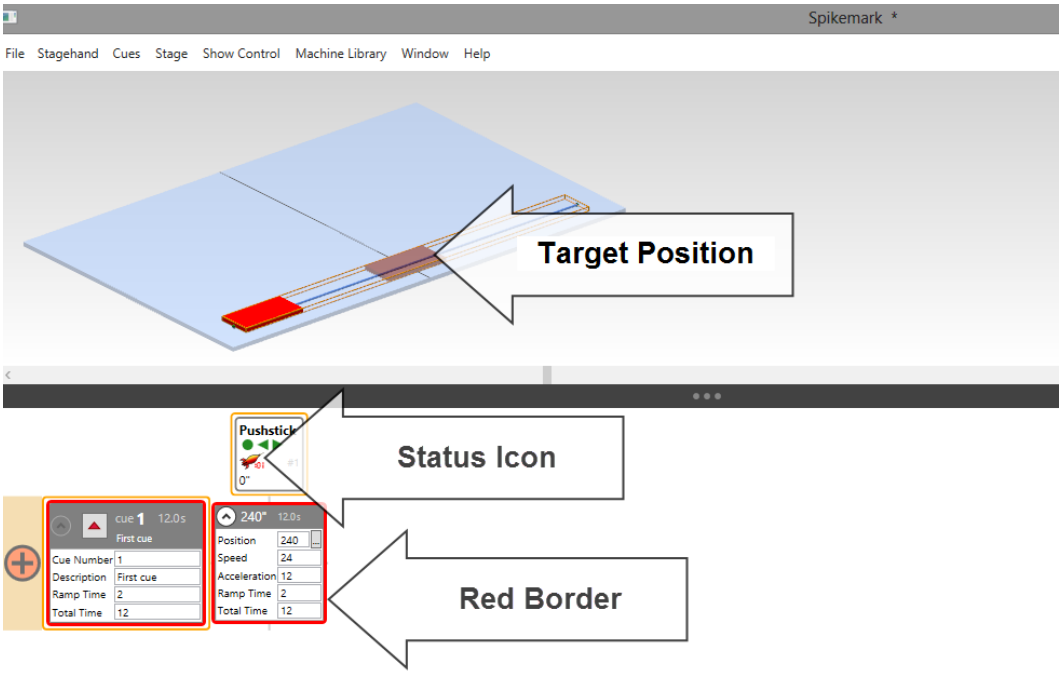


Figure 26

7. Press the **Run** button to watch the *Pushstick* wagon move across the stage. It turns green while running (Figure 27) and then blue when it completes the cue (Figure 28). That's it, you wrote your first automation cue!

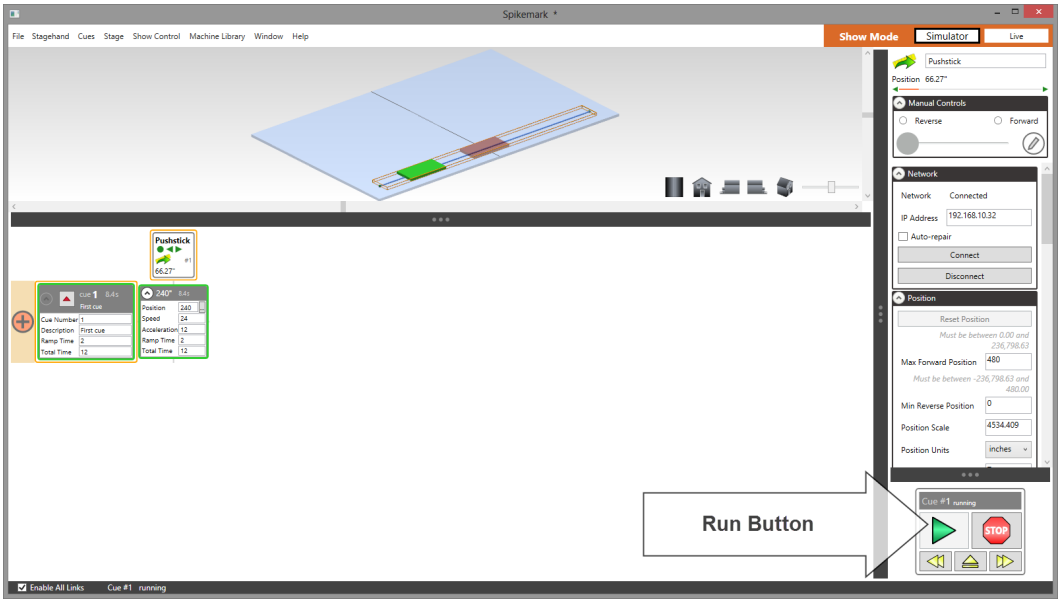


Figure 27

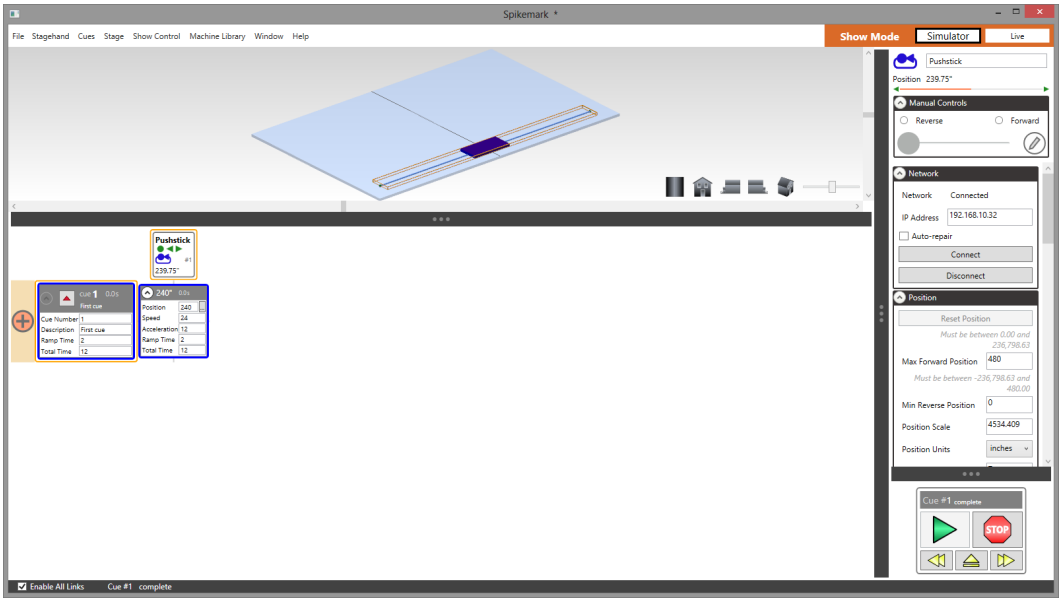


Figure 28

Adding a Second Machine to the Show

Now that you see how easy it is to move some scenery around, you probably want to add more to the show. In this section, we are going to add a turntable using our pre-configured [Revolver machine](http://creativeconners.com/products/shop-a-la-carte/machinery/revolver) (http://creativeconners.com/products/shop-a-la-carte/machinery/revolver).

1. The easiest way to add another automated piece of scenery to a show is to use Spikemark's **Machine Library**. Click on the **Machine Library** menu (Figure 29).

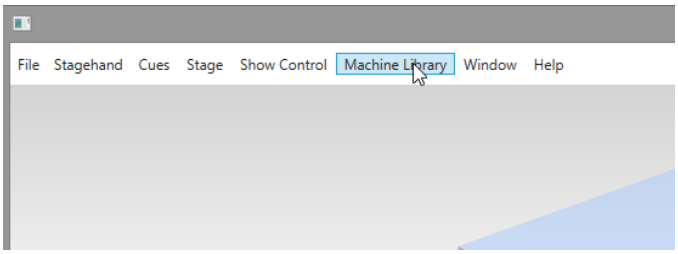


Figure 29

2. The **Machine Library** slides down into view (Figure 30). The **Machine Library** is pre-loaded with all the stock machinery from Creative Conners: **Pushstick** for deck wagons, **Revolver** for turntables, and **Curtain Call** for traveler tracks.

Figure 30

3. We are going to add a turntable to our show. Drag a **Revolver** from the **Machine Library** and drop it either on the **Stage Model Viewer** or the **Cue Grid** (Figure 31), either works fine.

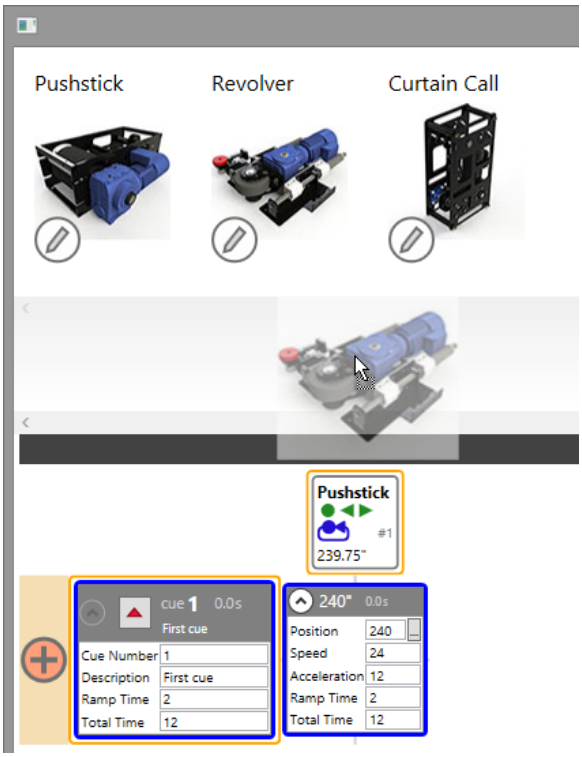


Figure 31

4. When you drop the **Revolver** into the show, the **Revolver Diameter** dialog box is displayed. Enter in a diameter of 20 (for twenty feet) and press **OK** (Figure 32).

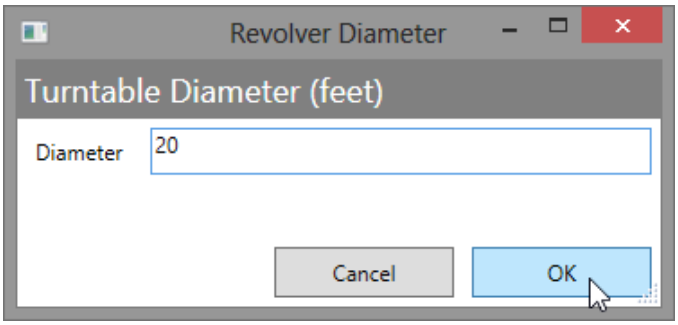


Figure 32

NOTE: Spikemark uses the diameter you enter to calculate an approximate Position Scale. Position Scale is used to run physical machinery, so you needn't worry about it if you only want to work in the Simulator.

5. We are finished with the **Machine Library** for now, so click the **Close Tab** to roll it up and out of the way (Figure 33).

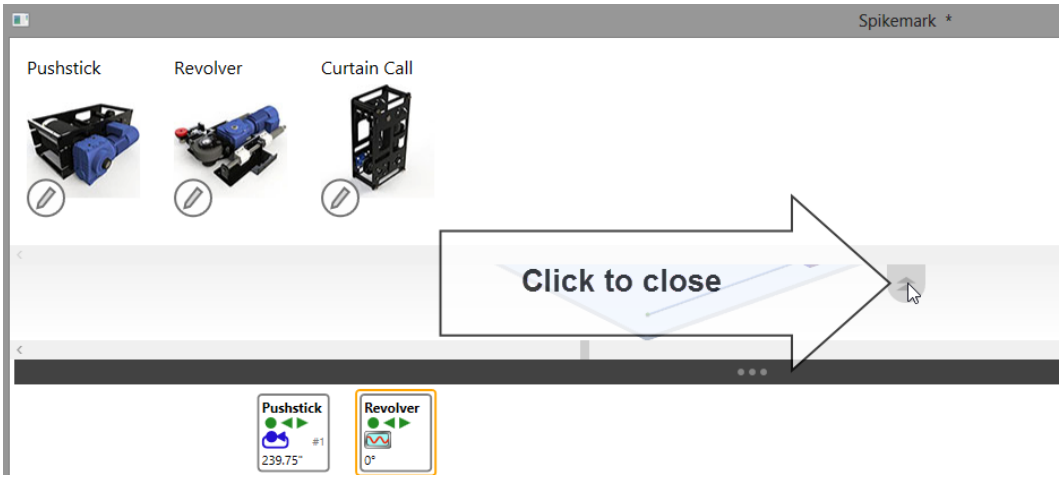


Figure 33

6. You can see that a 20’ diameter turntable has been placed on center line of the stage and a little bit upstage of the *Pushstick* wagon (Figure 34).

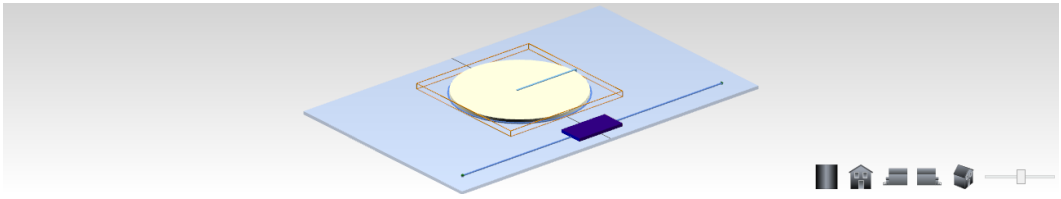


Figure 34

7. Before we move on to the next section and write some more cues, let’s take a second to rename our new turntable. Rather than use the name *Revolver*, let’s label it *Turntable*. Click on the name in the **Stagehand Editor** and change the name to *Turntable* and then press the Tab key to finish editing. You’ll see that the name updates in the **Cue Grid** to read *Turntable* (Figure 35).

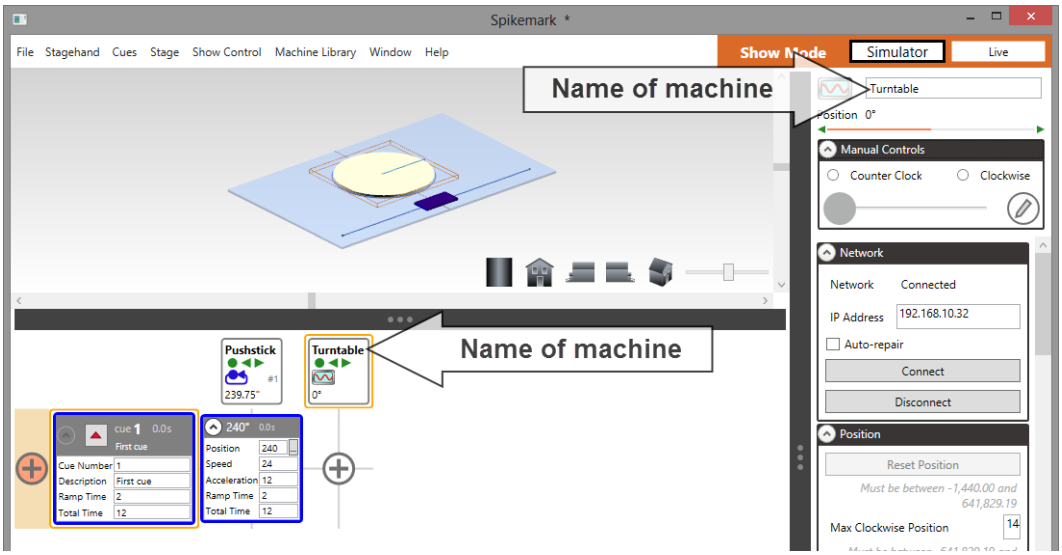


Figure 35

Writing Cues with Multiple Machines

Writing cues with two or more machines is just as easy as writing cues with a single machine. The following steps will guide you through the process, and along the way you will learn a couple of other handy features in Spikemark.

1. Let's add another cue to our simple show. From the **Cues** menu select **Add Cue…** (Figure 36)

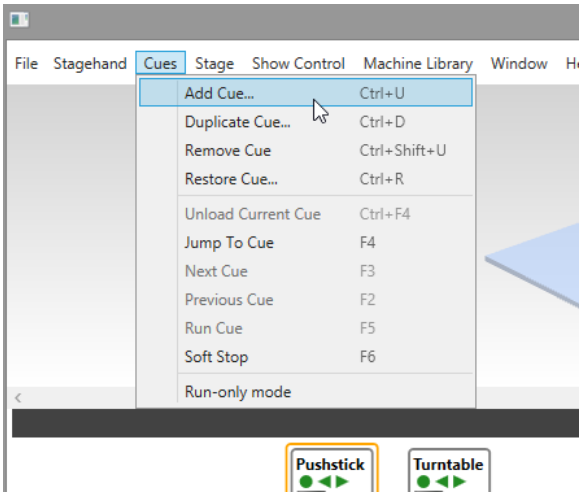


Figure 36

2. The **New Cue** dialog appears and suggests a **Cue Number** of 2, since that is the next highest whole number for a cue. Even though it was nice of Spikemark to offer a suggestion, let's type in "1.234" for a cue number and then press OK (Figure 37).

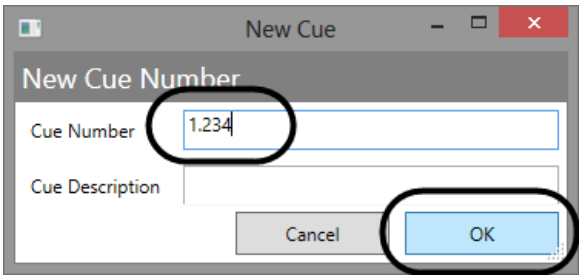


Figure 37

3. A new row is added to the **Cue Grid** for *Cue 1.234* (Figure 38).

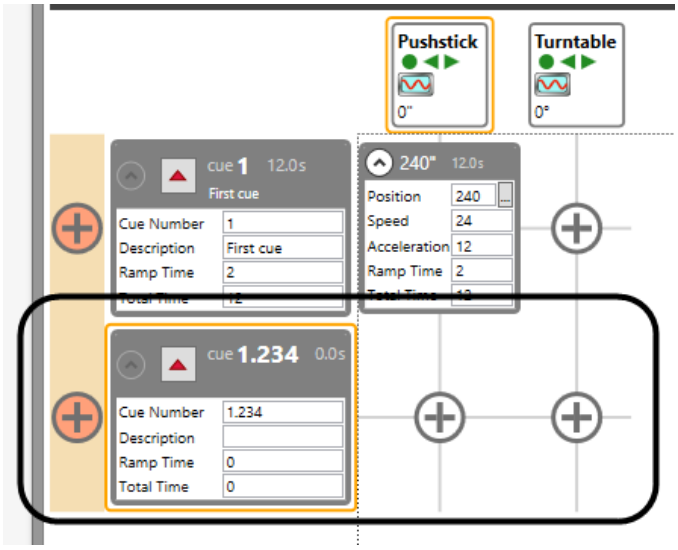


Figure 38

4. Before we add the *Turntable* into *Cue 1.234*, let's manually run it into the position that we wish to record in the cue so we can set the position visually. Select the *Turntable* either by clicking on it in the **Cue Grid** or the **Stage Model Viewer** (Figure 39).

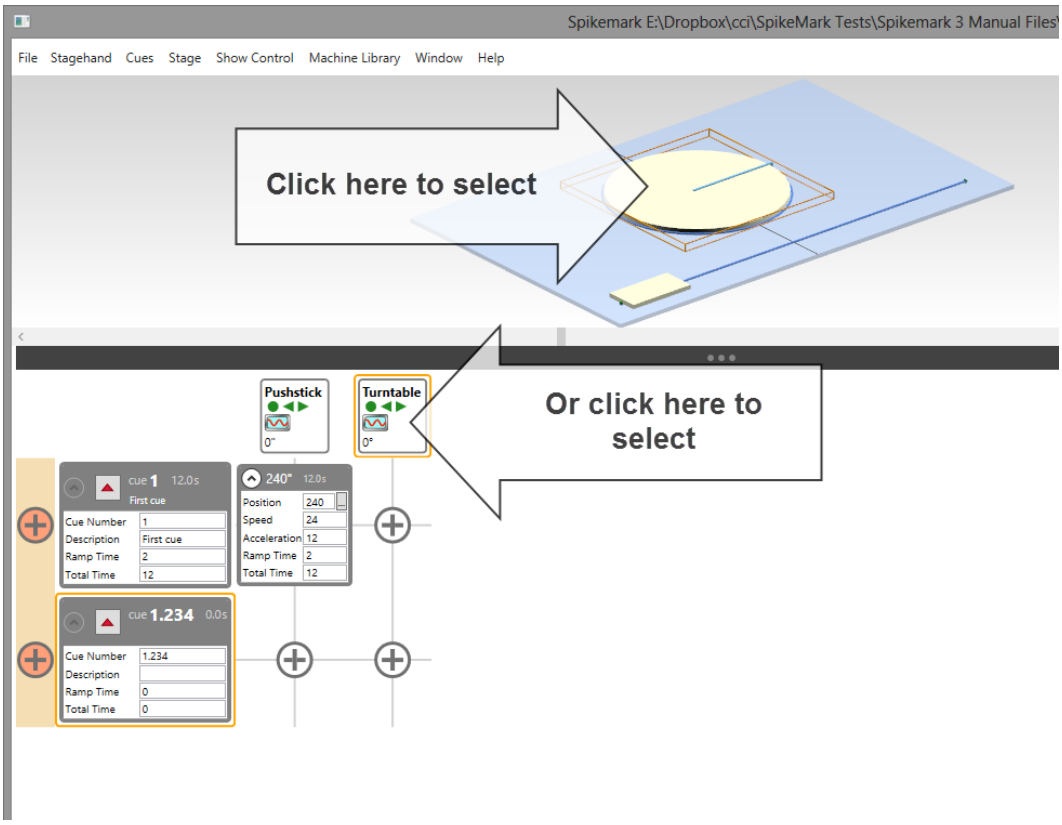


Figure 39

5. When a machine is selected, it will have an orange border around it in the **Cue Grid** and the **Stage Model Viewer**. Once selected, that machine's properties will be displayed in the **Stagehand Editor** on the right side of the screen (Figure 40).

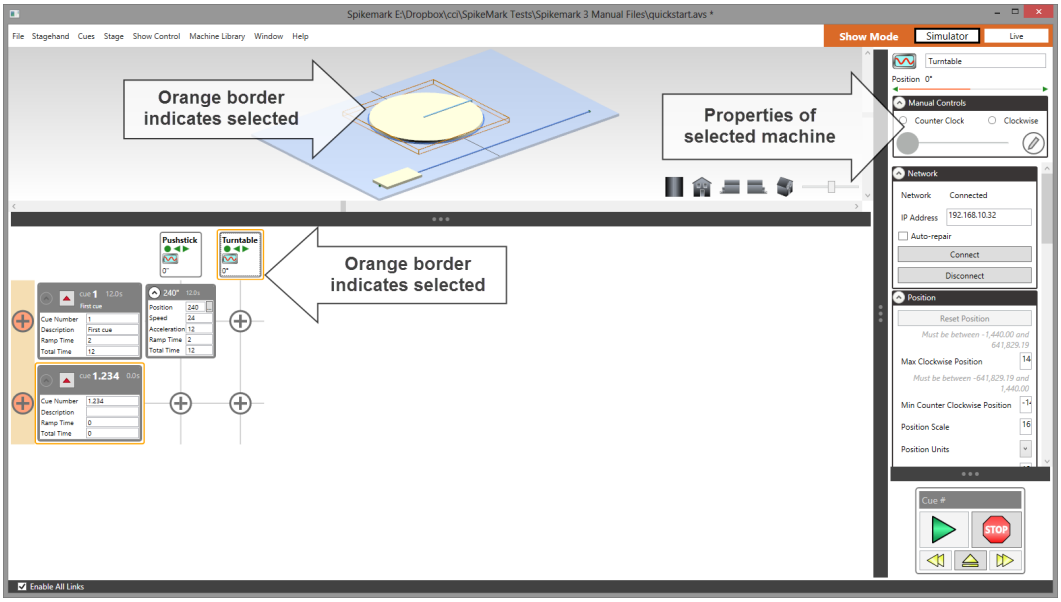


Figure 40

6. With the *Turntable* selected, we are going to run it clockwise until it is pointing downstage. Click on the **Clockwise** radio button in the **Manual Controls** section of the **Stagehand Editor** (Figure 41).

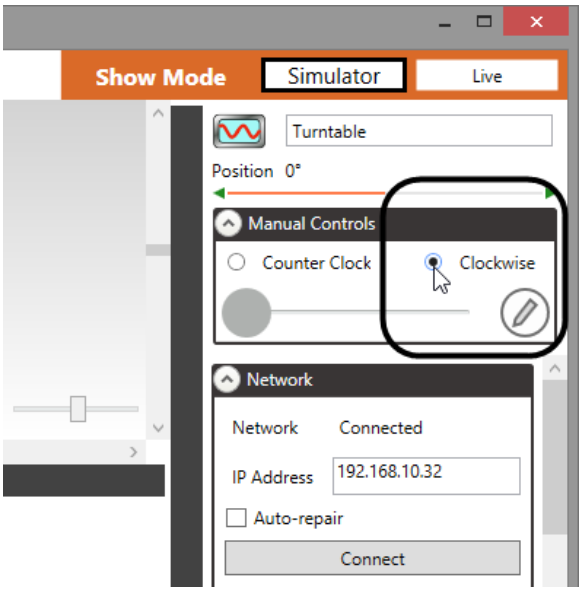


Figure 41

7. Click and drag the **Speed Slider** in the **Manual Controls** section of the **Stagehand Editor** (Figure 42). You will see the *Turntable* start moving in the **Stage Model Viewer** and the **Position** numbers start to increase in both the **Stagehand Editor** and the **Cue Grid**.

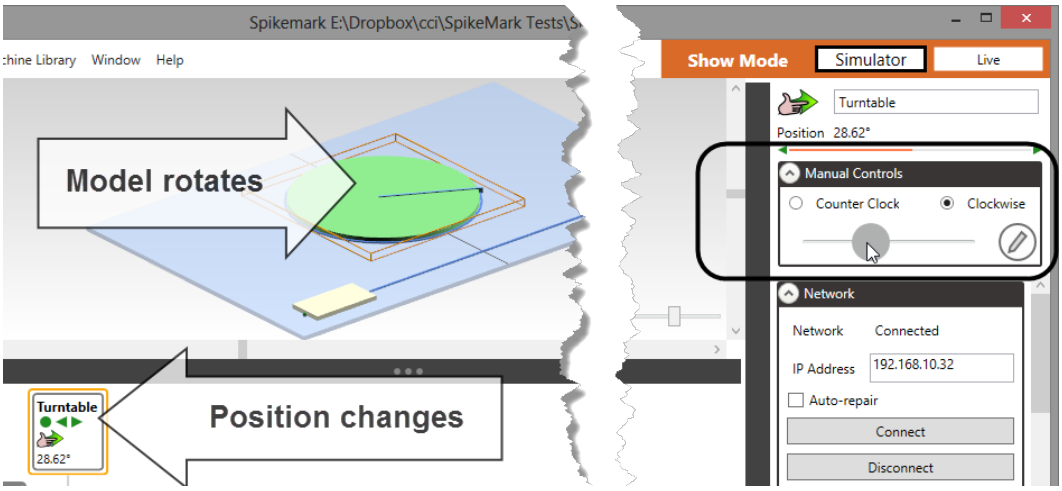


Figure 42

8. When the *Turntable* is pointed downstage with a **Position** of roughly 90 degrees, lower the speed slider back down and release it to stop the *Turntable* (Figure 43).

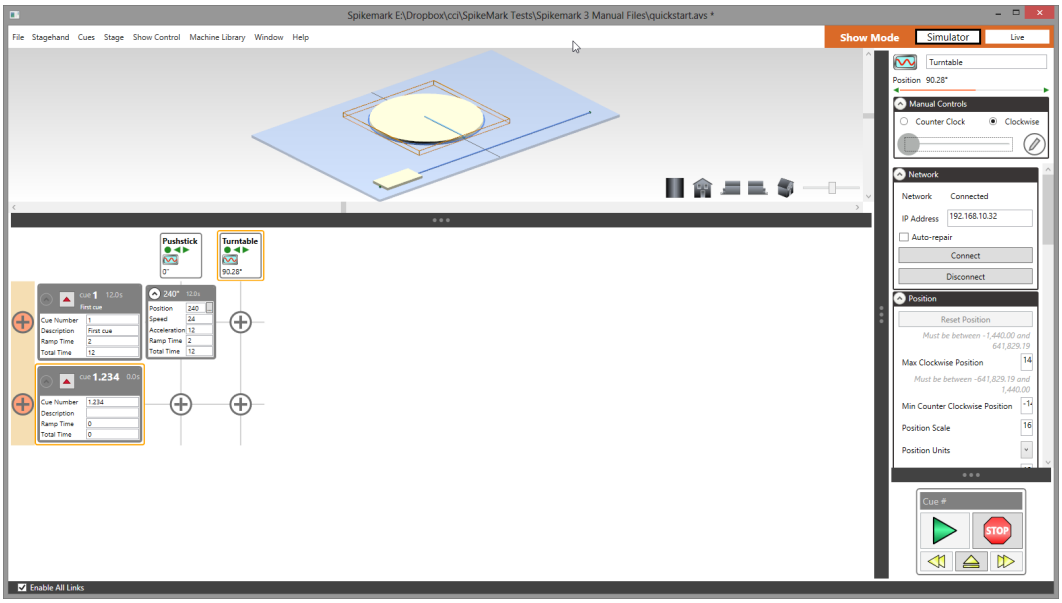


Figure 43

9. With the *Turntable* in a position we want to record, add a *Turntable* movement to *Cue 1.234* by clicking on the "+" button at the intersection of *Turntable* and *Cue 1.234* (Figure 44). Notice that Spikemark records the current position of the *Turntable* as the target position for the movement (Figure 45). This makes it easy to position all your scenery on stage manually, and then record the scene as a cue. You are free to tweak the **Position** anytime, but for now let's just leave it.

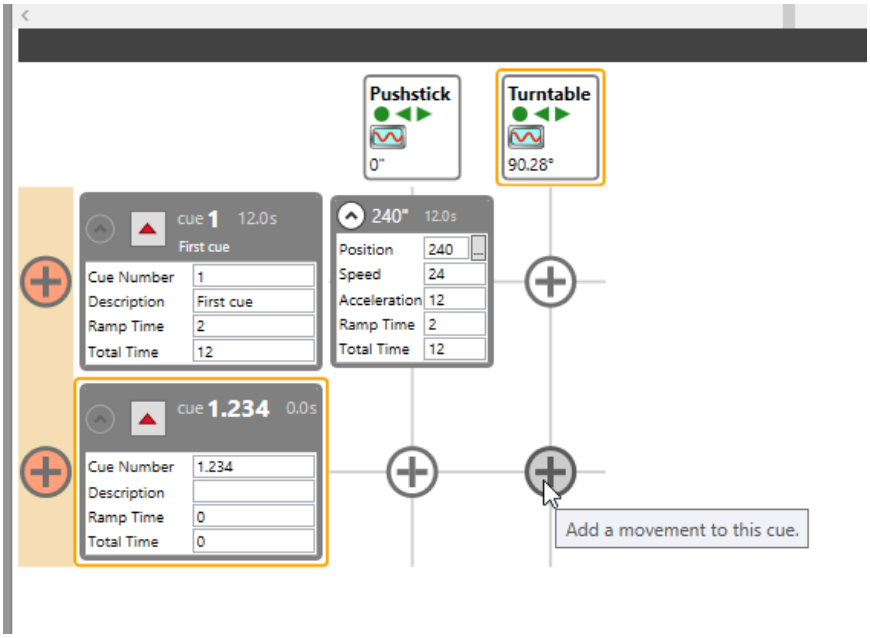


Figure 44

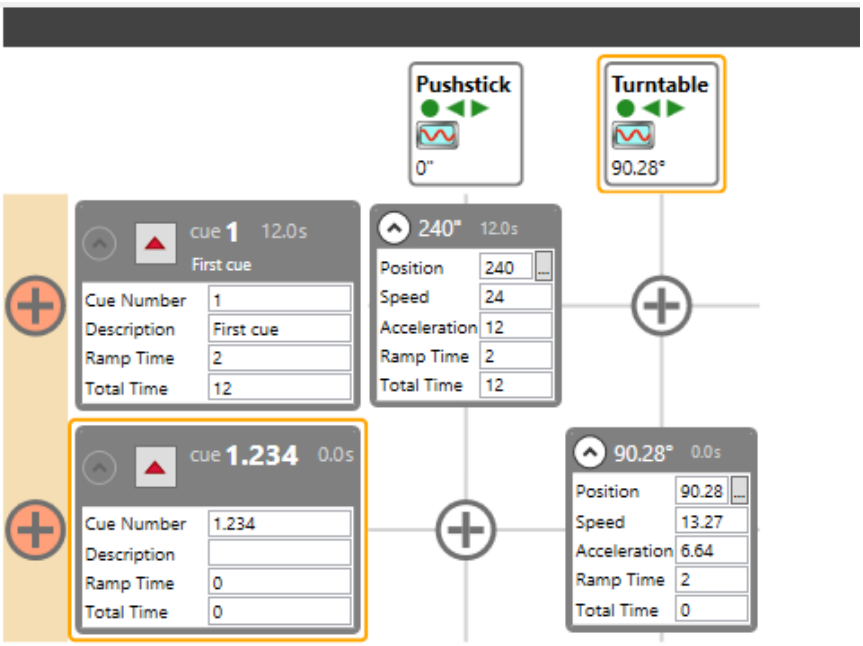


Figure 45

10. Let's add another cue. From the **Cue** menu select **Add Cue…**
11. Enter "2" for a **Cue Number**.
12. In *Cue 2*, we want both the *Pushstick* and the *Turntable* to move so click the "+" buttons for both movements in *Cue 2*.
13. Enter a **Position** value of 0 for the *Pushstick* (Figure 46).
14. Enter a **Position** value of 0 for the *Turntable* (Figure 46).

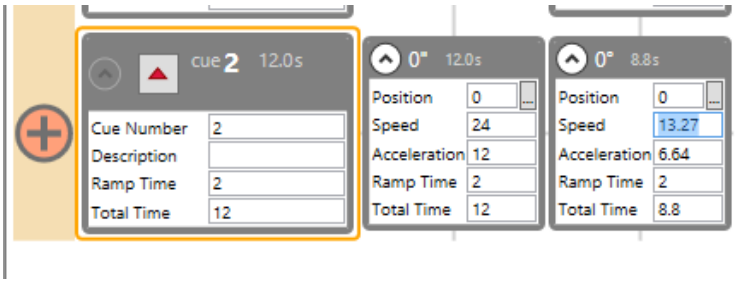


Figure 46

15. Rather than adjust the **Speed** and **Acceleration** values, let's have Spikemark do the math to figure out how fast to move both machines if we want the cue to last 11 seconds. Enter 2 into the **Ramp Time** field and 11 into the **Total Time** field of *Cue 2* (Figure 47).

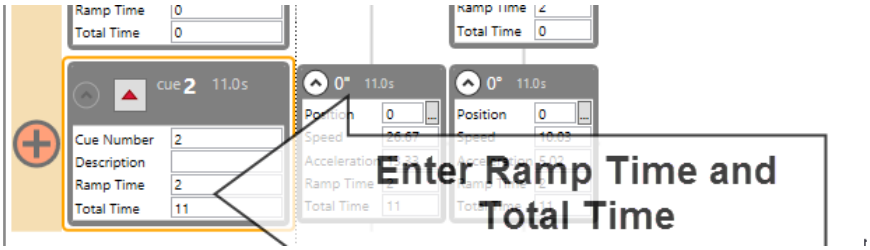


Figure 47

NOTE: Speed and Acceleration values are given in units/sec and units/sec/sec (e.g. degrees/second or inches/second). Ramp Time and Total Time values are given in seconds. You can fill in either set of values for a cue or movement, but don't both Speed/Acceleration and Ramp Time/Total Time.

To illustrate the point, imagine requesting that a pilot fly a plane from New York to Los Angeles in 1 hour at a speed of 60mph. Only one of those requests can be honored because of the laws of physics. You can request travel duration, or a speed, but not both.

16. Press the **Load this cue** button in the *Cue 2* panel (Figure 48).

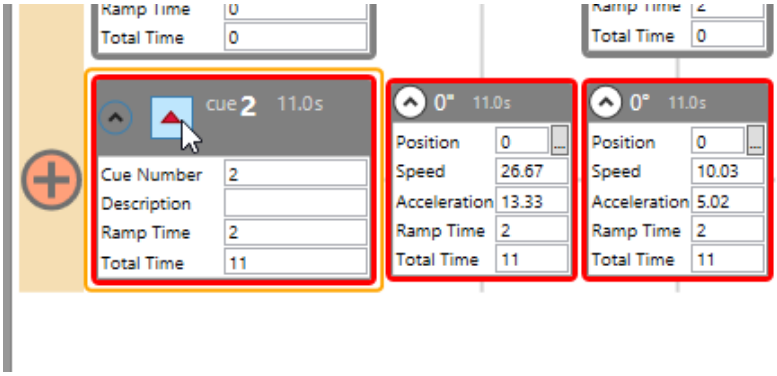


Figure 48

17. Press the **Run** button and watch the cue execute (Figure 49). Notice how both the *Pushstick* and the *Turntable* start and end at the same time since we specified a **Total Time** for the entire cue.

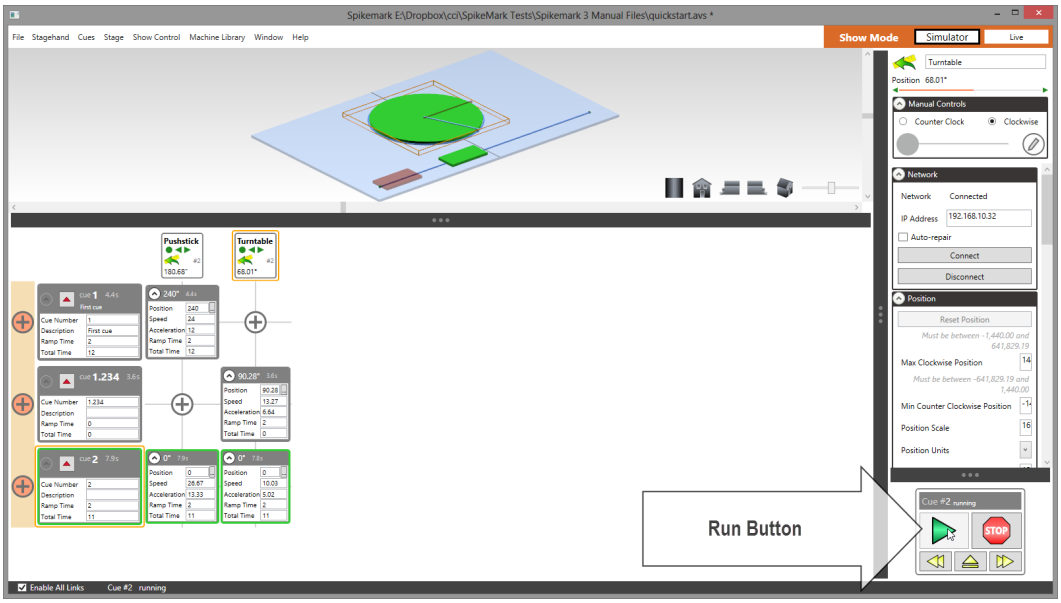


Figure 49

In the next section we'll take a look at using **Spikes** to define positions before moving real machinery in **Live Mode**.

Using Spikes

So far we've been typing in position values for every movement. That is a valid technique, but using **Spikes** to define positions offers some valuable benefits.

Traditionally, when you need to push scenery out on stage during a scene change you stick a piece of spike tape on the floor so you know when the scenery is at the proper position. You grab a marker and label the spike tape with a useful name like "Act II Parlor". Bits of tape on the floor end up being a much more practical way of placing scenery at the correct spot on stage rather than keeping a pocket notebook of measurements and running out with a tape measure during every scene shift to place the set. Besides avoiding the obvious embarrassment of standing on stage with a tape measure scratching your head, it also makes changing the position simpler. If you have to change a position during tech rehearsal, you lift the tape and move it instead of searching through a notebook and finding all the places in the show where you need to adjust the measurement.

Spikes behave like virtual spike tape inside Spikemark. You can set positions and label them. You can program scenery to move to those positions in every cue that shares the spike, and if you find that you need to change the **Spike** position during rehearsal, just edit the value in one place and all the cues are automatically updated with the new position.

If we think about the positions we've used so far in our tiny show, each motor has two positions that we could turn into **Spikes**.

Pushstick:

- Offstage = 0”
- Center stage = 240”

Turntable:

- Pointing stage left = 0 degrees
- Pointing downstage = 90 degrees

To create **Spikes** for the *Pushstick*:

1. Click on the “…” button next to the **Position** field in the *Pushstick* **Movement** panel for *Cue 1* (Figure 50).

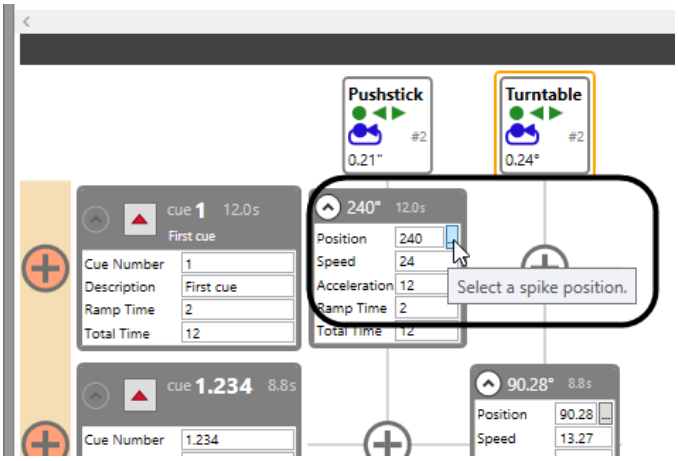


Figure 50

2. The **Spikes** window will appear. Add two spikes to the *Pushstick* by clicking the “+” button twice (Figure 51).

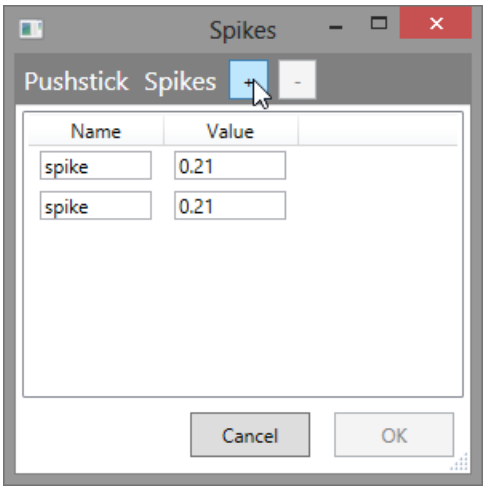


Figure 51

3. Enter the **Names** and **Values** as shown below (Figure 52).

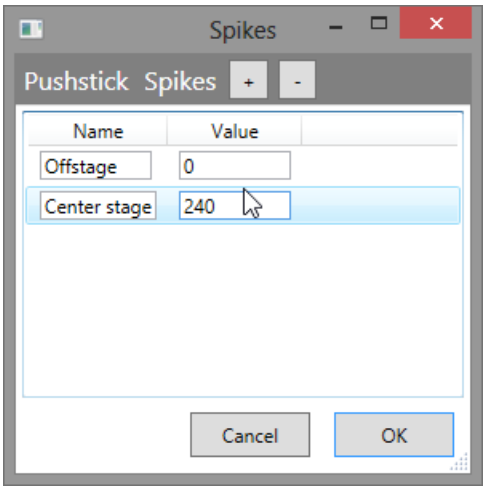


Figure 52

4. Select the *Center Stage* **Spike** to use it in *Cue 1* (Figure 53).
5. Press **OK** (Figure 53).

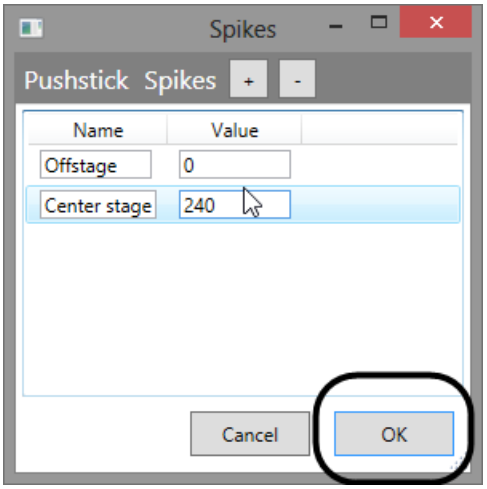


Figure 53

6. Notice that the *Pushstick* **Movement** for *Cue 1* shows the *Center Stage* **Spike** name for the target position (Figure 54).

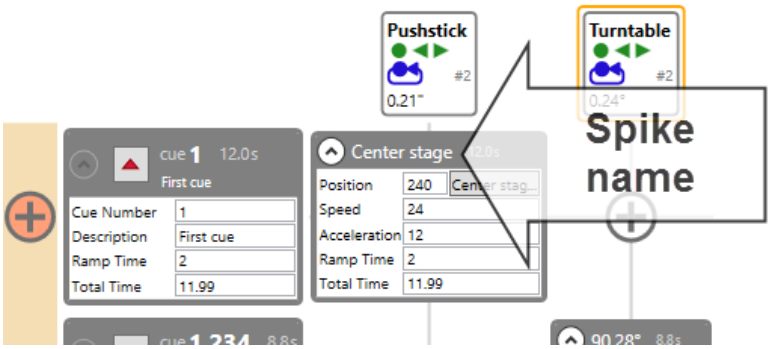


Figure 54

7. In the *Pushstick Cue 2* **Movement**, click the "**…**" button next to the **Position** field.
8. Select the *Offstage* **Spike**.
9. Press OK.
10. Notice that the *Pushstick* **Movement** for *Cue 2* shows the *Offstage* **Spike** name for the target position.

To create **Spikes** for the *Turntable*:

1. Click on the "**…**" button next to the **Position** field in the *Turntable* **Movement** panel for *Cue 1.234* (Figure 55).

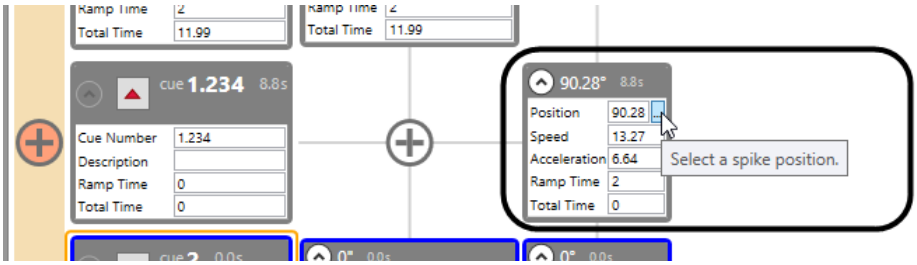


Figure 55

2. The *Spikes* window will appear. Add two spikes to the *Turntable* by clicking the "+" button twice.

3. Enter in the **Names** and **Values** as shown below (Figure 56).

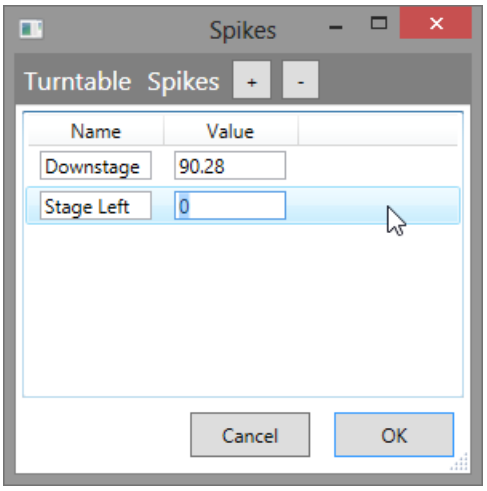


Figure 56

4. Select the *Downstage* **Spike** to use it in *Cue 1.234* (Figure 57).

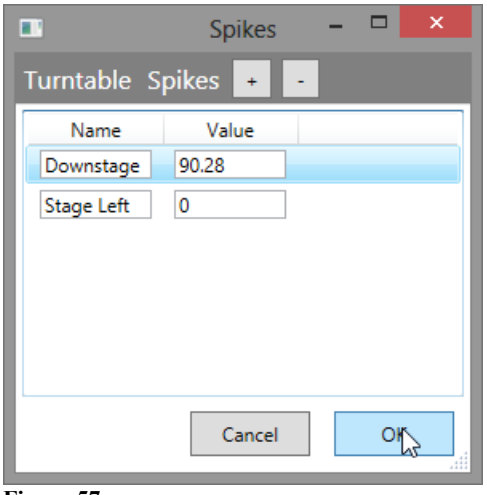


Figure 57

5. Press **OK**.

6. Notice that the *Turntable* **Movement** for *Cue 1.234* shows the *Downstage* **Spike** name for the target position (Figure 58).

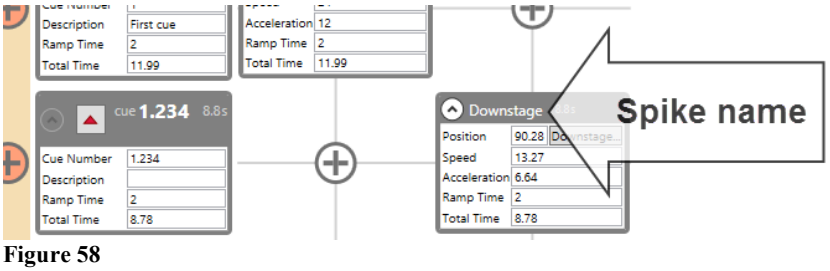


Figure 58

7. In the *Turntable Cue 2* **Movement**, click the "**…**" button next to the **Position** field.

8. Select the *Stage Left* **Spike**.

9. Press **OK**.

10. Notice that the *Turntable* **Movement** for *Cue 2* shows the *Stage Left* **Spike** name for the target position.

With the **Spikes** defined, let's take a look at how easy it is to adjust a **Spike** position.

1. Press the "**…**" button next to the **Position** field for either of the *Turntable's* **Movements**.
2. Change the value for the *Downstage* **Spike** to 95 (Figure 59).

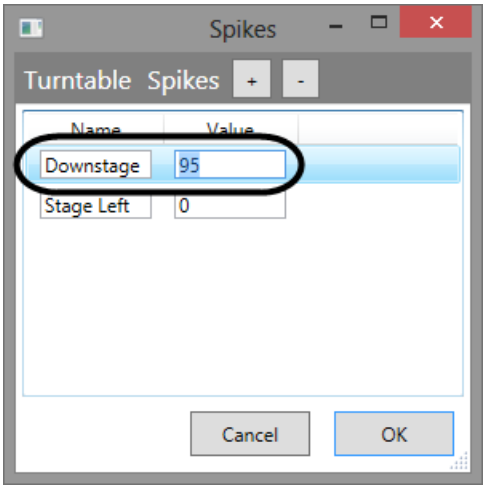


Figure 59

3. Press **OK**.
4. Notice that the position for the *Turntable* in *Cue 1.234* has been updated to reflect the new **Spike** value (Figure 60). If we had multiple cues referencing the *Downstage* **Spike**, all of them would be updated automatically.

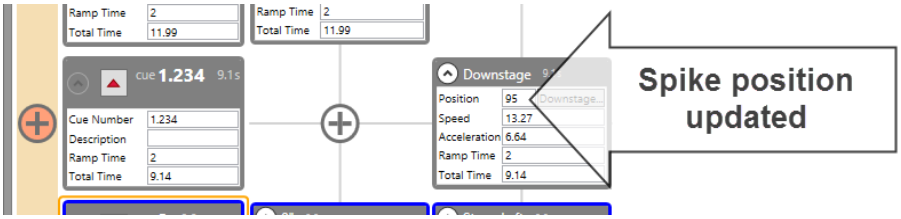


Figure 60

TIP: A handy way to pre-program cues for a show is:

- **Write Spikes for all the cue positions of each scenic element in a script.**
- **Assign approximate positions for each Spike, but don't worry too much about accuracy.**
- **Write cues in the Simulator and go over the rough cues with Stage Management.**

- **During load-in, run each motor to the correct positions on stage and adjust the values for each Spike to match the correct physical position.**

All of your cues that are already written will be updated with accurate positions and you'll head into tech with a great starting point for your show.

That wraps up our **Simulator** Quick Start. If you have some Stagehand controllers and real machinery (Pushstick, Revolver, Curtain Call, or something you made yourself) read on through the next section to learn how to setup your network and make some scenery move. If you want to continue experimenting in the **Simulator**, read through some of the other chapters on more advanced cue topics like **Cue Links** on page 134 and **Dataton WATCHOUT Integration** on page 193.

Moving Real Machinery with Spikemark

The best way to learn to use Spikemark with real machinery is to start using it, so let's jump in and get something moving!

Setting Up Physical Connections

Before anything will work, all of your cables should be connected, as illustrated in Figure 61.

NOTE: Make sure to set the physical limits on your motor[s] before running the motor with Spikemark.

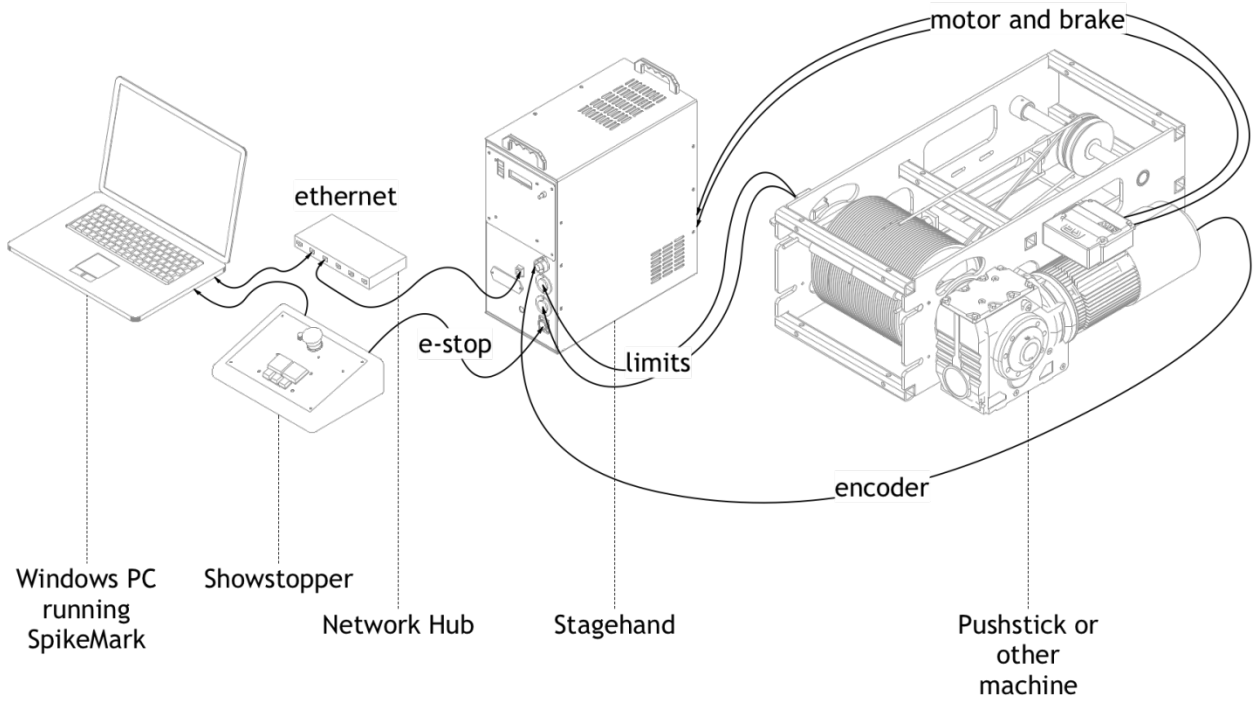


Figure 61

Setting Up a Network Connection (in Windows 8)

Once everything is plugged in, we need to test that the connections work. We'll start by confirming that the Ethernet link works. Spikemark and Stagehands communicate through static IP addressing. This means that your computer and all Stagehands need to have unique addresses. Much like a postman delivering letters to homes with address numbers nailed to the mailbox, Spikemark sends its commands to Stagehands with unique address numbers.

IP addresses are made up of four segments separated by a period (for example, "192.168.10.20"). The only trick to setting up a static IP network with Spikemark and Stagehands is that the first three segments should always match. The last segment number needs to be unique (Figure 62).

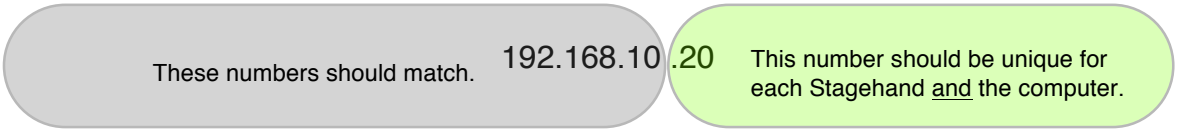


Figure 62

Setting the IP Address of Your Computer

1. Press the **Windows Key** + **X** and select **Control Panel** from the pop-up menu (Figure 63).

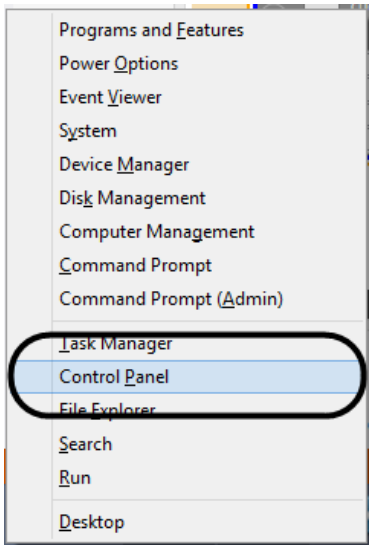


Figure 63

2. Click on the **Network and Internet** link (Figure 64).

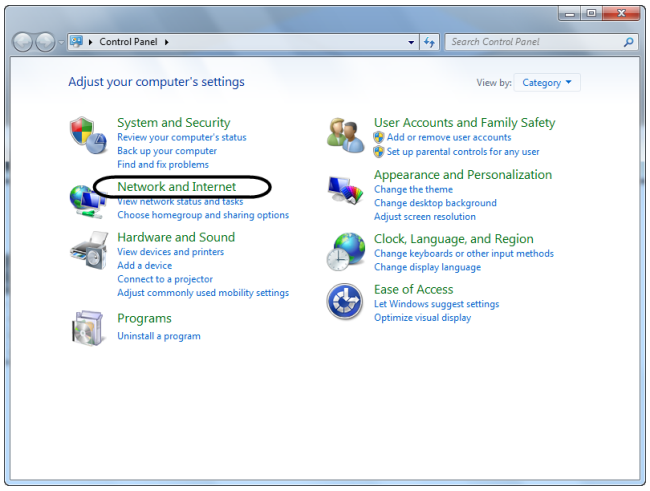


Figure 64

3. Click on the **Network and Sharing Center** link (Figure 65).

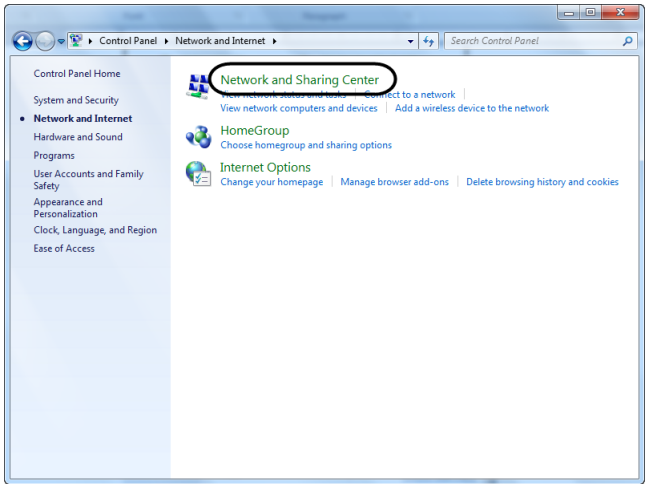


Figure 65

4. Click on the **Local Area Connection** link (Figure 66).

   Note: if you don't see **Local Area Connection**, click on **Change adapter settings**, select your Ethernet-wired connection and select **Change settings of this connection** in the top menu.

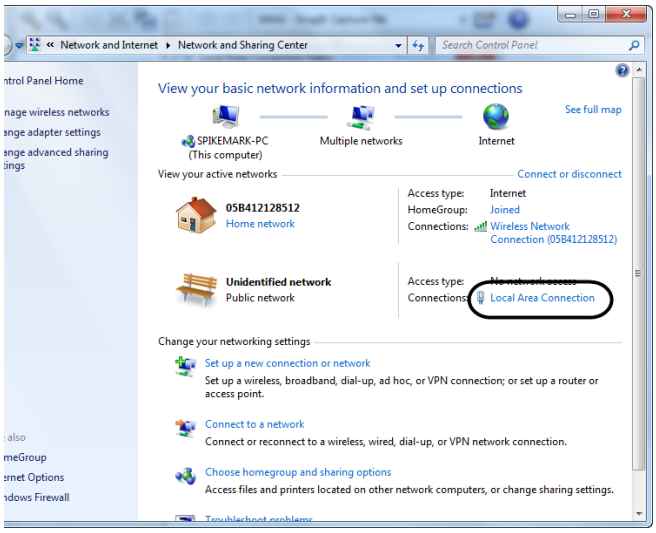

Network and Internet
Network and Sharing Center
Search Control Panel
View your basic network information and set up connections
See full map
SPIKEMARK-PC
(This computer)
Multiple networks
Internet
View your active networks
Connect or disconnect
05B412128512
Home network
Access type: Internet
HomeGroup: Joined
Connections: Wireless Network Connection (05B412128512)
Unidentified network
Public network
Access type:
Connections: Local Area Connection
Change your networking settings
Set up a new connection or network
Set up a wireless, broadband, dial-up, ad hoc, or VPN connection; or set up a router or access point.
Connect to a network
Connect or reconnect to a wireless, wired, dial-up, or VPN network connection.
Choose homegroup and sharing options
Access files and printers located on other network computers, or change sharing settings.

Figure 66

5. In the **Local Area Connection Status** dialog box, click the **Properties** button (Figure 67).

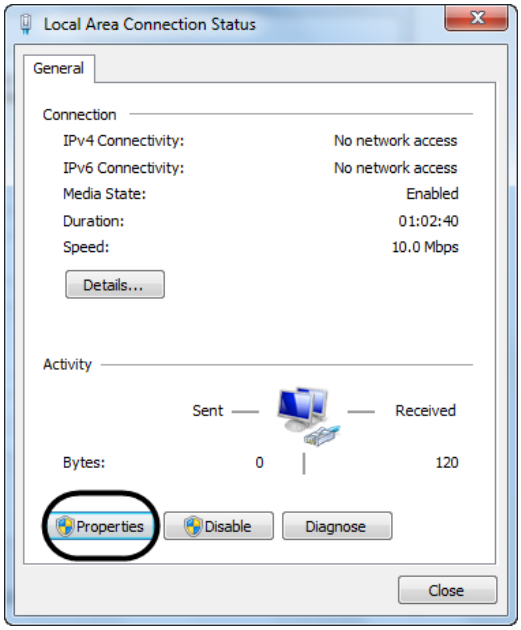


Figure 67

6. Highlight **Internet Protocol Version 4 (TCP/IPv4)** and then click **Properties** (Figure 68).

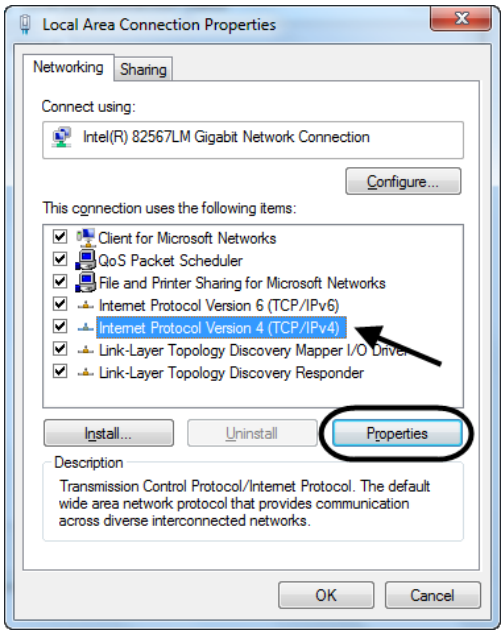


Figure 68

7. In the **Internet Protocol Version 4 (TCP/IPv4) Properties** dialog box, select **Use the following IP address** field and then enter the correct **IP Address** correct **Subnet mask** of **255.255.0.0** (Figure 69).

NOTE: The first 3 segments of the IP address must be <u>shared</u> by your computer and any Stagehand(s). The last segment of the IP address MUST be unique to each Stagehand and your computer. The IP address of your Stagehand(s) is displayed on the face panel.

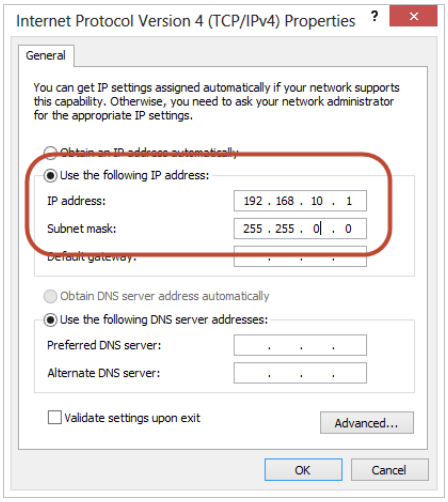

Internet Protocol Version 4 (TCP/IPv4) Properties
General
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Use the following IP address:
IP address:
192 . 168 . 10 . 1
Subnet mask:
255 . 255 . 0 . 0
Obtain DNS server address automatically
Use the following DNS server addresses:
Preferred DNS server:
Alternate DNS server:
Validate settings upon exit
Advanced...
OK
Cancel

Figure 69

8. Click **OK** to accept the new **IP address**.
9. Click **Close** in the **Local Area Connection Properties** dialog box.
10. Click **Close** again in the **Local Area Connection Status** dialog box.

Adding Machinery to the Show

To start a new show, launch Spikemark. When Spikemark first opens, it begins a new show with one Pushstick winch and one cue. This is reasonable start for a very simple show. Chances are good, though, that you want to add more, or different, machines to your show (and certainly more cues!). There are two ways to add a new machine to your show.

- Add a new machine to the show from the **Machine Library**. The **Machine Library** stores templates of machines with all of the setup parameters defined so you don't have to spend time configuring motor parameters each time you start a new show. Spikemark comes with all of Creative Conners' machines pre-defined in the **Machine Library**. Over time, you can add your own machines to the **Machine Library**. Using the **Machine Library** is the easiest way to get started.

- Add a new blank machine with the **Add Motor** or **Add Stagehand FX** command. If you have a new custom machine that needs to be configured from the ground up, use the **Add Motor** or **Add Stagehand FX** command. Using these commands requires that you configure some setup parameters before writing cues. See *The Stagehand Motor* (page 65) and *The Stagehand FX* (page 103) for detailed configuration information.

Add a Pre-Configured Machine from the Machine Library

1. Click or touch the **Machine Library** menu (Figure 70).

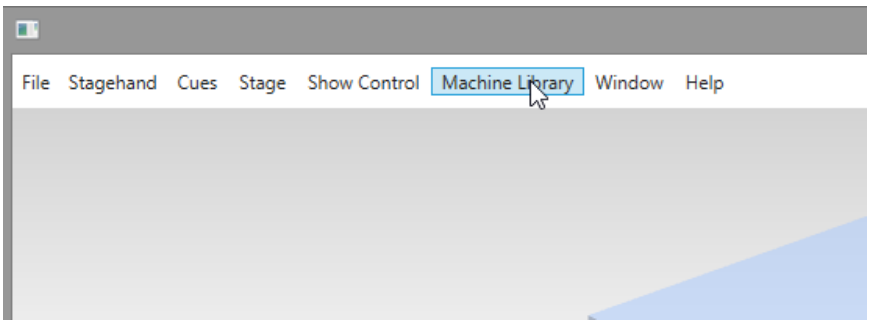


Figure 70

2. The **Machine Library** panel will slide down from the top of the main window (Figure 71).

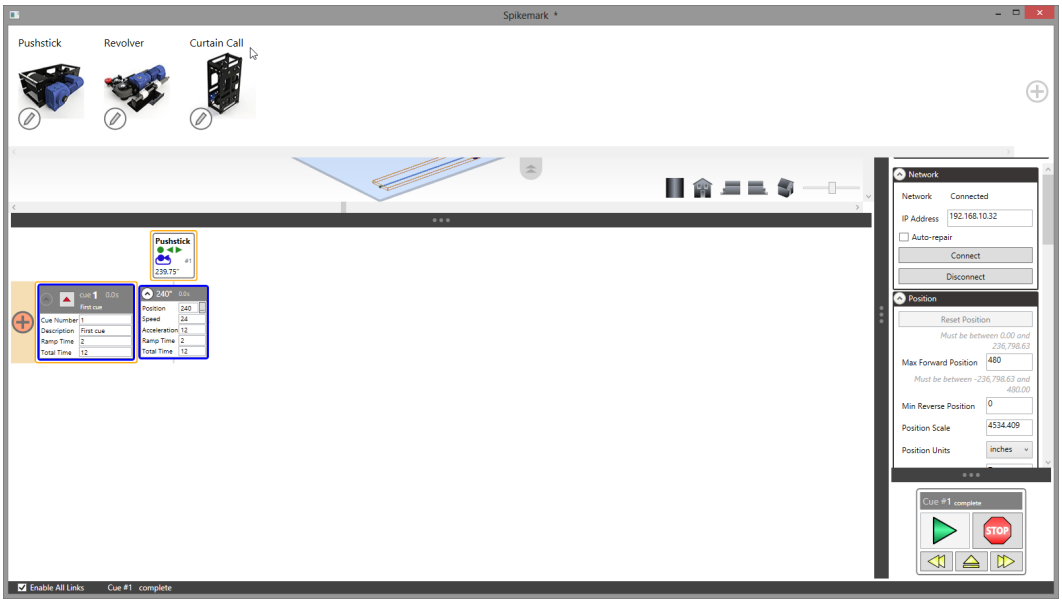


Figure 71

3. Drag one of the machines from the **Machine Library** panel and drop it on either the **Stage Model Viewer** or the **Cue Grid** (Figure 72).

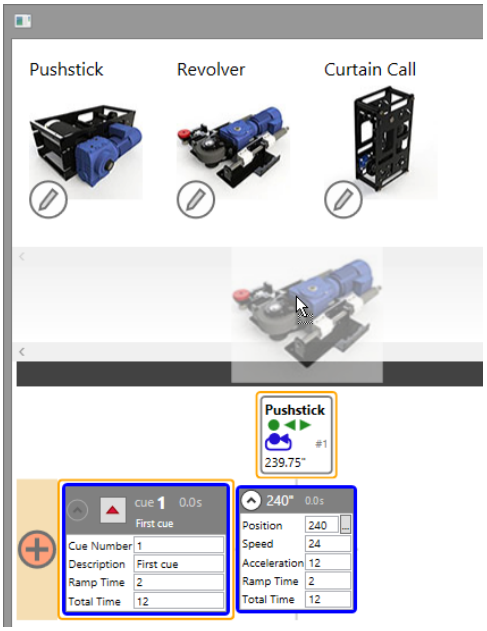


Figure 72

Add a New Blank Machine

There are a couple of ways to add a new, blank machine to your show.

1. From the **Stagehand** menu, select **Add Motor** or **Add Stagehand FX** (Figure 73).

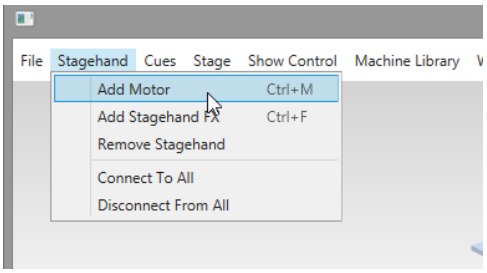


Figure 73

2. Right-click, or long-press, on the **Cue Grid** and select **Add Motor** or **Add Stagehand FX** from the pop-up menu (Figure 74).

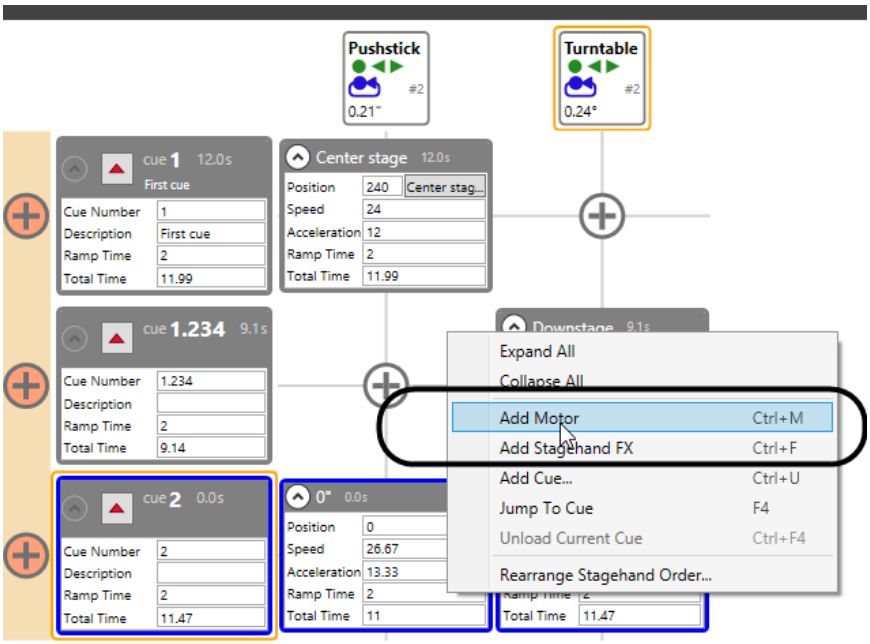


Figure 74

NOTE: If you add a new blank Motor to the show, you will need to configure Position Scale, Max Speed, Default Speed, Default Acceleration, and the PID Tuning before moving on. See The Stagehand Motor starting on pg. 65 for details.

Removing a Machine from the Show

It is good to know how to get rid of a machine from the show in case you add one by accident, or perhaps you find out that the automation budget has been cut before you start tech rehearsal. There are two ways to remove a machine (either a Motor or Stagehand FX) from the show.

1. Select the machine by clicking on it in either the **Cue Grid** or the **Stage Model Viewer** (Figure 75).

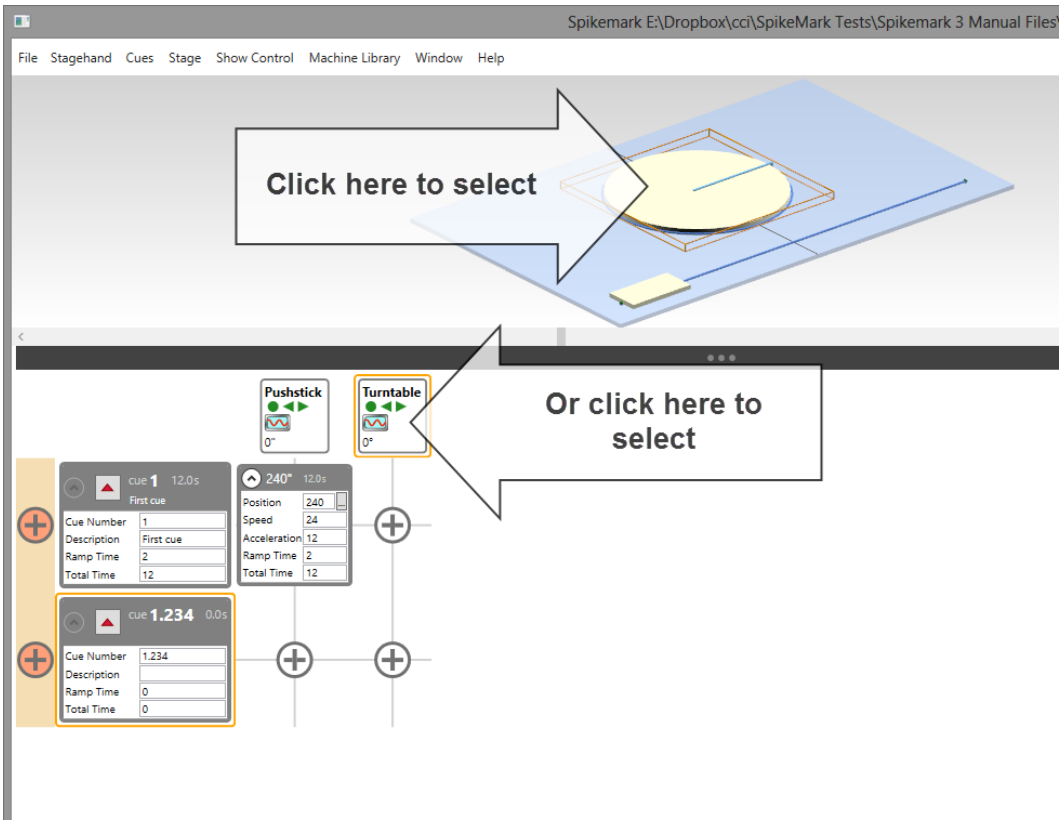


Figure 75

2. With the machine that you wish to remove selected, choose **Remove Stagehand** from the **Stagehand** Menu (Figure 76).

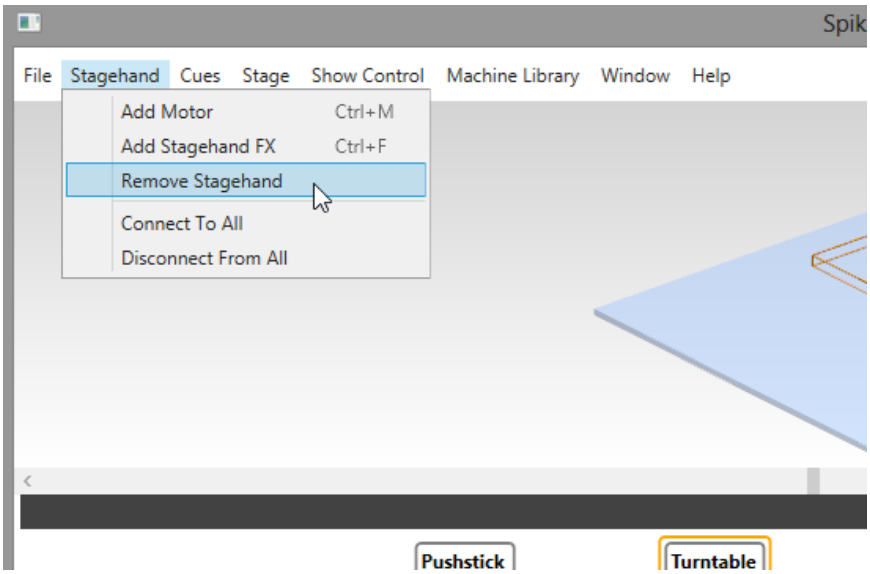


Figure 76

-OR-

1. Right-click on the **Cue Grid** header of the machine you wish to remove, and select **Delete [machine name]** from the pop-up menu (Figure 77).

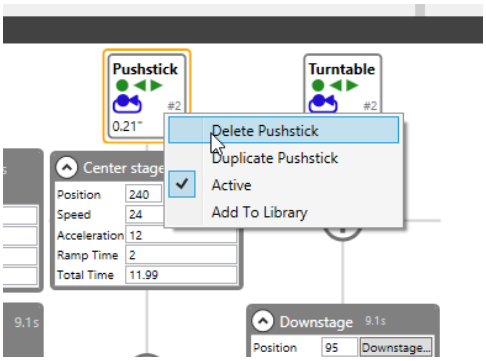


Figure 77

Establishing a Network Connection to your Machine

With the network address configured on your computer and a machine or two, we're ready to establish a connection to a Stagehand.

1. With Spikemark Open, select **Live** from the **Show Mode** switch (Figure 78).

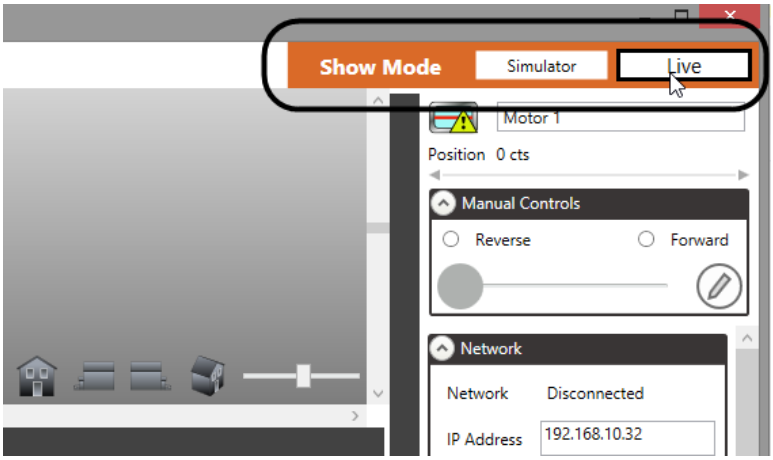


Figure 78

2. Select the motor or FX you wish to connect to by clicking on it in the **Cue Grid**. An orange highlight will appear around the selected motor (Figure 79).

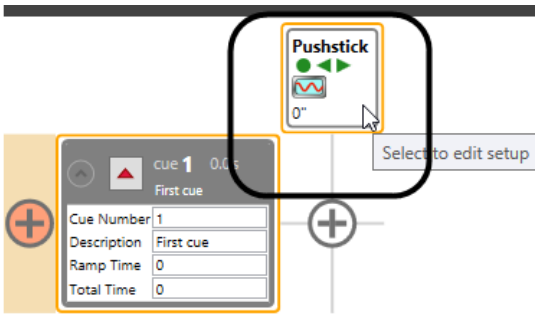


Figure 79

3. In the **Network** section of the **Stagehand Editing Pane** on the right side of the screen, enter the **IP Address** of your Stagehand Motor or FX and click **Connect** (Figure 80).
   *In this step, make sure you copy the IP Address exactly as it appears on the face panel of the **Stagehand** LCD display.*

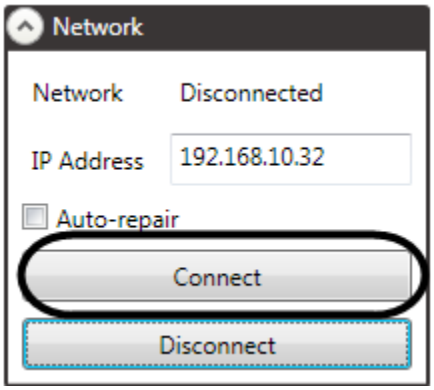


Figure 80

4. The Network status should then change from **Disconnected** to **Connected** (Figure 81).
   If there's a network error, check your IP settings on the computer and Stagehand and then check your cabling.

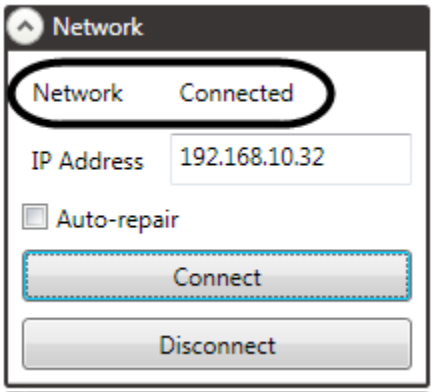


Figure 81

Confirming Encoder Feedback

1. Mark the motor's physical position on the stage with a piece of spike tape.

2. With the motor still selected in the **Cue Grid**, find the **Manual Controls** in the top portion of the **Stagehand Editing Pane** (Figure 82).

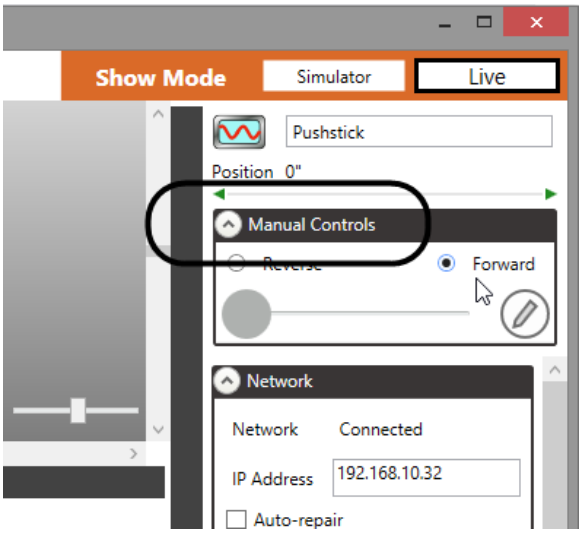


Figure 82

3. Select the **Forward** direction (Figure 83)

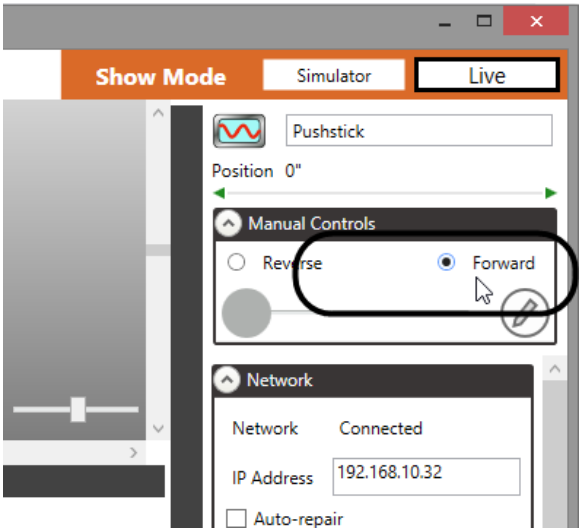


Figure 83

4. Make sure that there are no obstructions in the motor's path, and slowly start increasing the speed slider (Figure 84).
   You have to hold the mouse button down while the motor is moving. If you release the mouse button, the motor will stop. It's a dead-man switch.

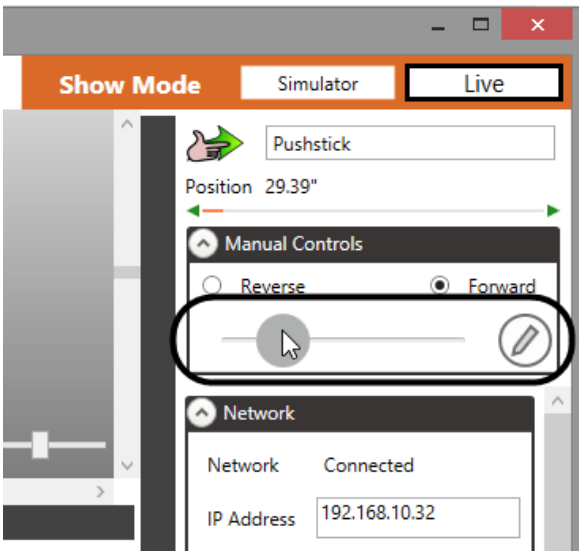


Figure 84

5. Confirm that the position number at the top of the **Stagehand Editing Pane** is increasing as the motor moves forward (Figure 85).

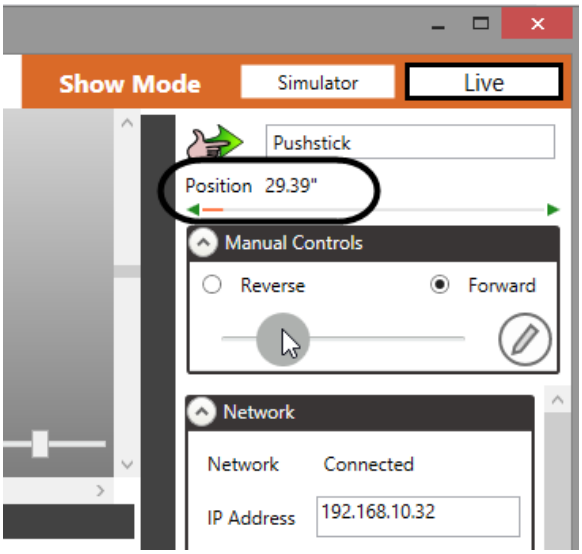


Figure 85

NOTE: The forward direction of the motor has to have increasing encoder counts. If the encoder counts are decreasing while the motor is moving forward, you need to fix the polarity of the motor. To fix the polarity of the motor, open up the motor plug and swap any two of the power wires or change the encoder wiring to reverse the encoder's polarity. (See NOTE, page 70).

DO NOT MOVE THE GROUND WIRE.

Setting Soft Limits

To prevent you from writing cues that attempt to move machinery beyond the physical limits of your space, Spikemark uses the **Max Fwd Position** and **Min Rev Position** configuration parameters to limit target positions in cues. This pair of parameters are referred to as **Soft Limits**. Machines from the **Machine Library** are inserted into the show with some "best guess" **Soft Limits**. Before writing cues and moving in **Live** mode, we should edit the **Soft Limits** to make sure that are set properly.

1. Using the **Manual Controls** in the **Stagehand Editing Pane**, drive the motor in the **Reverse** direction until it travels as far as you feel comfortable.

2. Click the **Reset Position** button in the **Stagehand Editing Pane** (Figure 86).

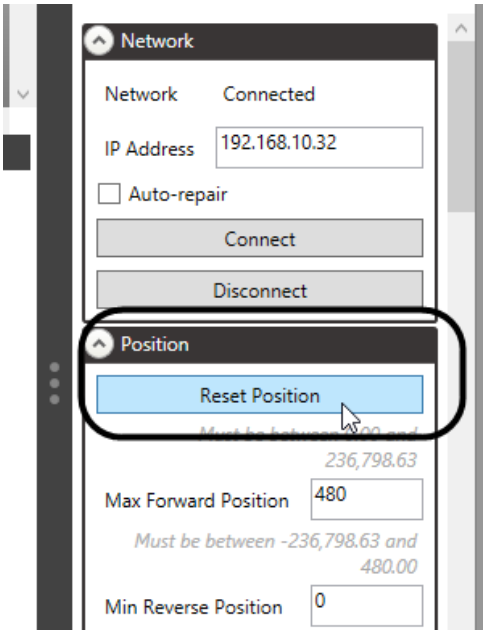


Figure 86

3. Enter "0" in the **Reset Position** dialog box. Press **OK** (Figure 87).

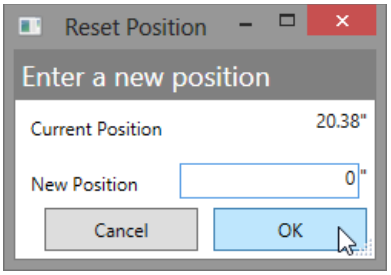


Figure 87

4. Place a piece of spike tape on the stage floor to mark the motor's physical position at the "0" location.

5. In the **Position** section of the **Stagehand Editing Pane**, enter “0” for the **Min Rev Position** (Figure 88).
   This will prevent you from accidentally writing cues less than 0.

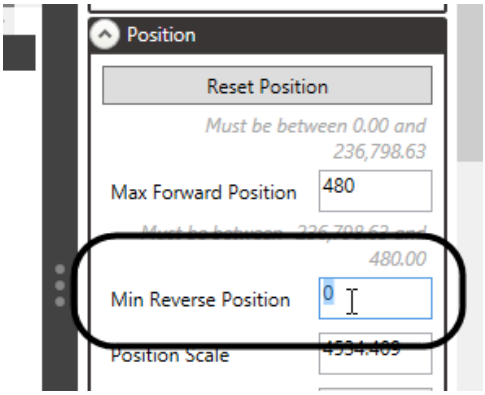


Figure 88

6. Using the **Manual Controls** in the **Stagehand Editing Pane**, drive the motor in the **Forward** direction until it travels as far as you feel comfortable.

7. In the **Position** section of the **Stagehand Editing Pane**, enter your current motor position as the **Max Fwd Position** (Figure 89), which will prevent you from writing cues greater than the current motor position.

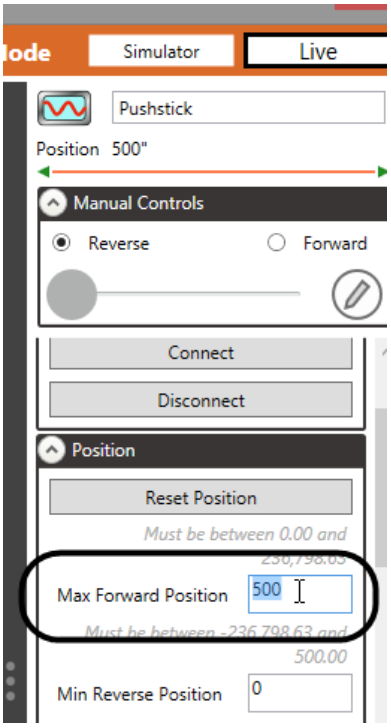


Figure 89

Writing Two Sample Cues

In the following steps, the first cue will send a motor to its Maximum Forward position (**Max Fwd Position**) and the second cue will send it back to its Minimum Reverse position (**Min Rev Position**). There's already a Cue 1 defined by default when you open Spikemark, so we'll add a Cue ".5," which will allow this cue to start *before* Cue 1.

1. Right-click in the **Cue Grid** and select **Add Cue**.
   *You could also press **Ctrl + U**, or from the **Cues** menu, click **Add Cue**.*

2. Enter a **Cue Number** and a **Cue Description** in the **New Cue** dialog box and click **OK** (Figure 90).
   Cue numbers can be any positive decimal value (1, 2.34, 1008.38972, etc.); Cue descriptions are labels you can use to help identify what the cue does and use to quickly search for a cue when running a show.

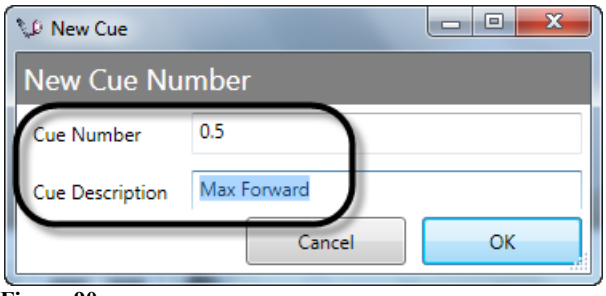


Figure 90

3. Add a movement for **Motor 1** in **Cue .5** by pressing the "**+**" button in the **Cue Grid** (Figure 91).

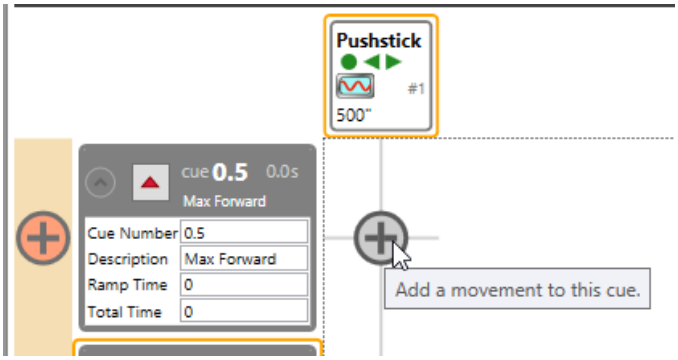


Figure 91

4. Change the **Position** number in the **Cue Grid** to match the value you entered as the **Max Forward Position.** (Figure 92).
   *In the example below, the value would be "500." This value is also held in the **Max Fwd Position** field in the **Position** section of the **Stagehand Editing Pane**.*

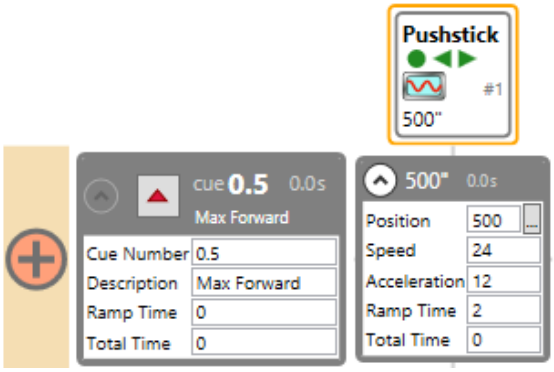


Figure 92

5. Leave the **Speed** and **Acceleration** values as their default values.

6. Now that we've added Cue .5, let's write another Cue. By default**, Cue 1** already exists. Enter its **Description** as "Min Reverse," then repeat steps 3 through 5 and enter a **Position** value of "0" (Figure 93).

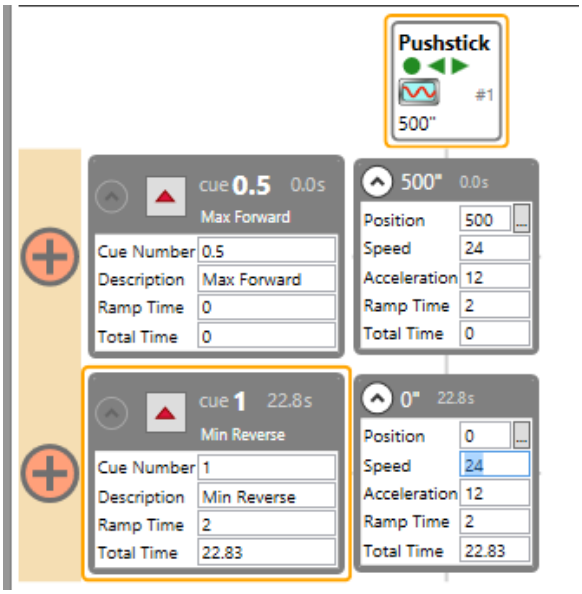


Figure 93

Running the Two Sample Cues

NOTE: Depending on your motor system, the cues may or may not work well at this point. We're going to run the cues with the default motor tuning parameters and hope for the best. If the default tuning isn't good enough, the motor won't complete the cue, it will just sit there struggling at very low speed to complete the cue. Don't worry if your cue doesn't run well yet, we'll fix it with motor tuning in the next section.

1. To load **Cue 1**, press the **Load Cue** button on the cue in the **Cue Grid** (Figure 94).

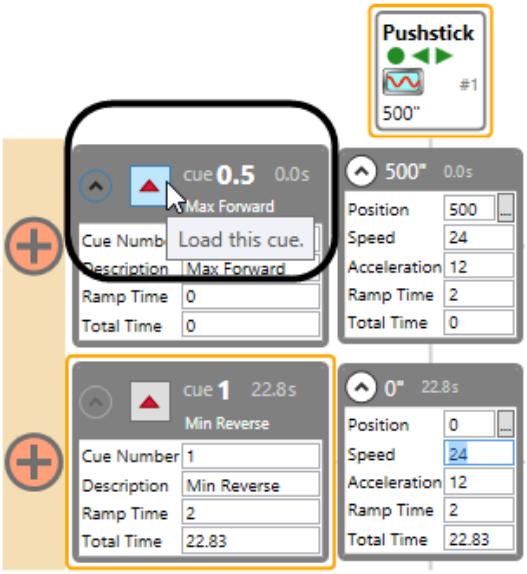


Figure 94

2. The border around the cue box will turn red, indicating that the cue is loaded (Figure 95).

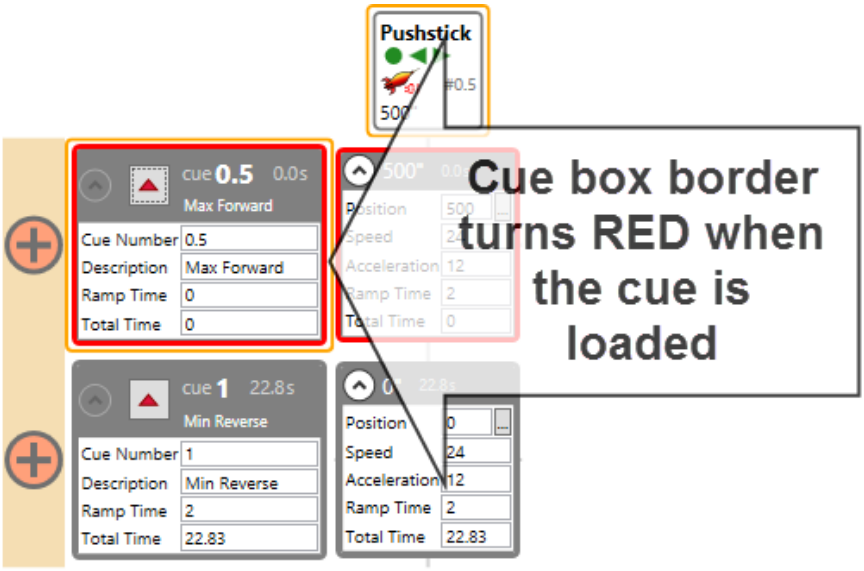


Figure 95

3. To run **Cue 1**, press the **Run Cue** button in the **Cue Controller** section on the right lower side of the screen (Figure 96).

Figure 96

4. As the cue is running, the border around the cue box in the **Cue Grid** turns green (Figure 97).

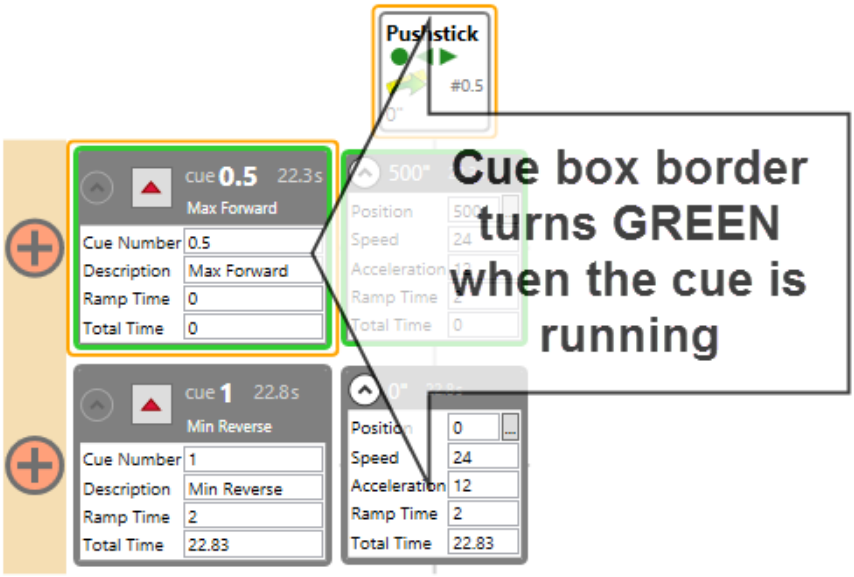


Figure 97

5. When the cue completes, the border around the cue in the **Cue Grid** turns blue (Figure 98).

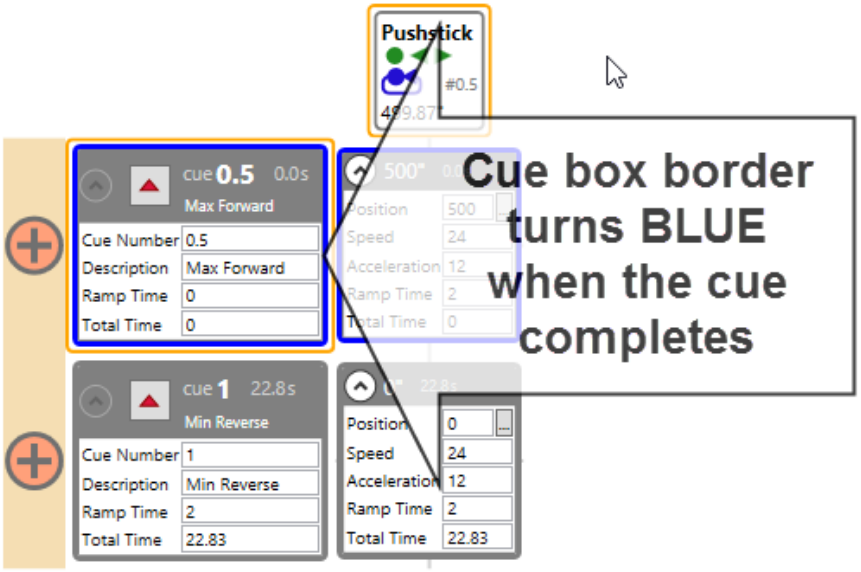


Figure 98

NOTE: If the border stays green, but the motor is no longer moving, this means that the motor tuning needs to be improved. Abort the cue by pressing the Emergency Stop button on your Showstopper, skip step 6 below and go to the next section: "Basic Tuning**," below.**

6. If the border around the cue box in the **Cue Grid** is blue and has properly completed, then run **Cue .5** by repeating steps 1 through 5 above.

Basic Tuning

If your first attempt at running cues didn't work and the motor never fully completed the cues, then the tuning parameters need to be adjusted. Tuning can be a complex issue, but we're going to take a practical approach in this section without getting bogged down in the details. For more information, see the "Motor Tuning" section on page 79.

1. In the **Stagehand Editing Pane** on the right side of the screen, scroll until you find the **Tuning** section.

2. Increase the **Proportional Gain** value by 1 (Figure 99).

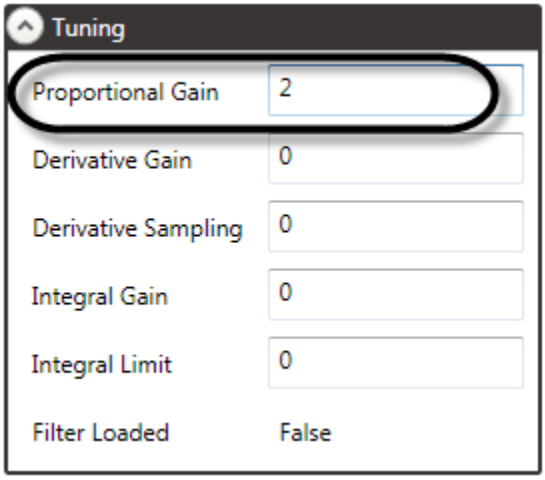


Figure 99

3. Run a Cue to move the motor to the opposite side of the stage by following the instructions in the previous section: “Running the Two Sample Cues.”

4. If the Cue now completes successfully, you are done. Otherwise repeat steps 1 through 3 until the motor repeatedly completes the cues successfully.

Conclusion

That wraps up our first look at moving real machines and real scenery around the stage. As you can see, Spikemark makes moving scenery simple. If you’ve made it this far in the manual you have a good grasp of how to use the Simulator, configure the network, connect to real machinery, and write some basic cues. In the following chapters we’ll dive deeper into the software and take a look at all the configuration options and more advanced methods for writing cues. Have fun!

Spikemark's Main Screen

The main screen of the Spikemark interface is composed of four areas (Figure 100, clockwise, from top left):

Stage Model Viewer: The upper left area, which contains and displays a 3-dimentional schematic representation of your theatre and motors.

Stagehand Editing Pane: The far right area, which allows setup parameters to be adjusted for a Stagehand Motor or Stagehand FX.

Cue Controller: The bottom right area, which mimics the functionality of a physical Showstopper, giving you the controls needed to run, skip, and stop cues.

Cue Grid: The lower center area, where cues can be added, removed, and edited.

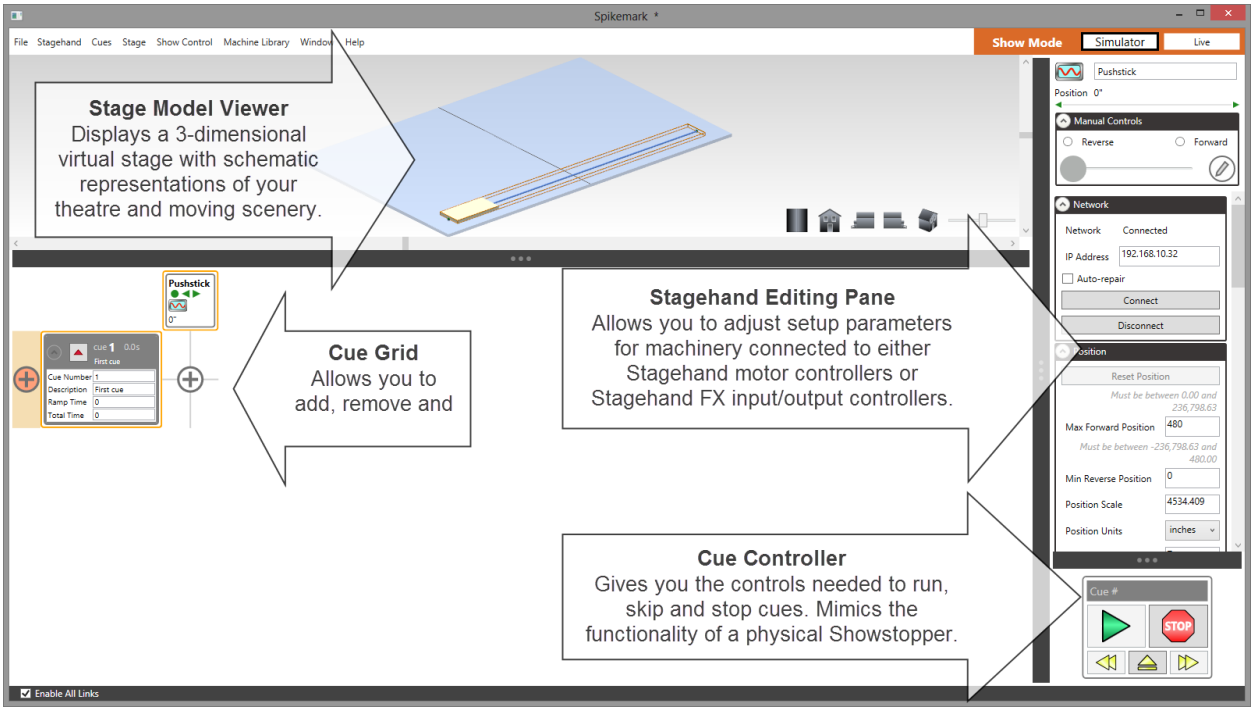


Figure 100

Splitter Panes allow you to adjust the size of the four sections of the Spikemark screen (the **Stage Model Viewer**, the **Motor** or **FX Editing Pane**, the **Cue Controller** and the **Cue Grid**). Click and drag on the **Splitter Panes** that separate each section (Figure 101). To fully collapse a pane, double-click on the **Splitter Pane**. To expand a collapsed pane, double-click again on the **Splitter Pane**.

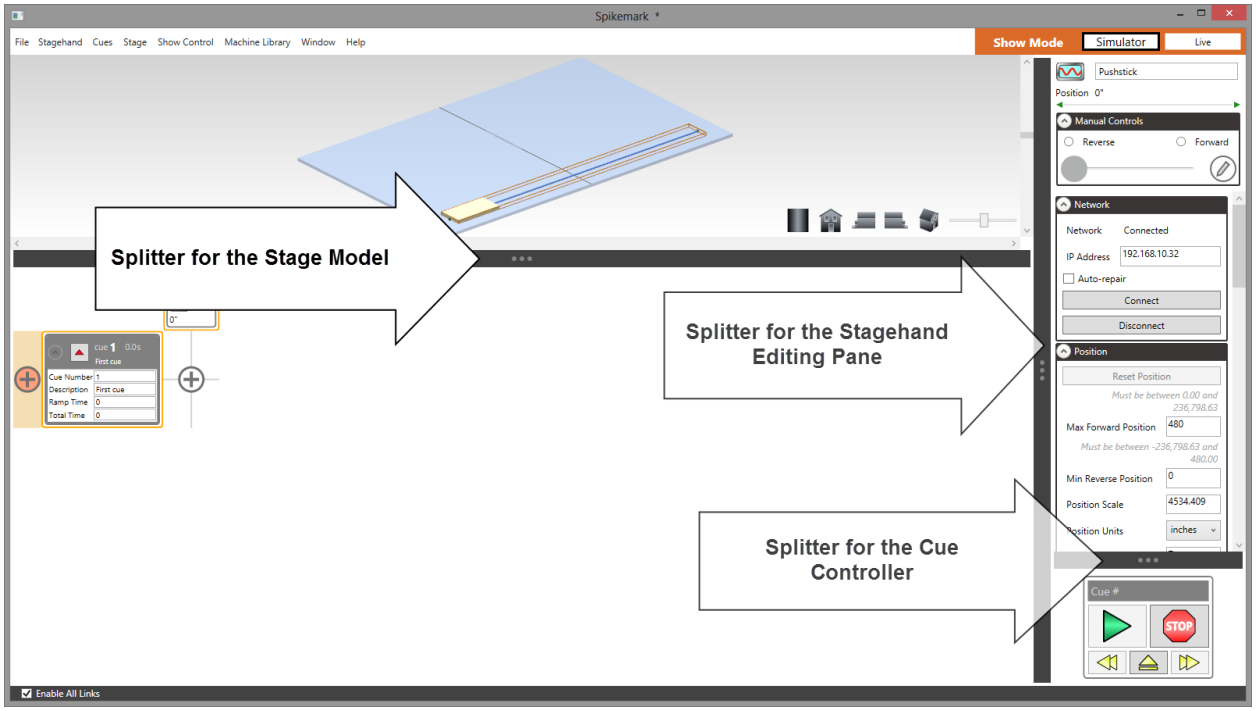


Figure 101

Simulator and Live Mode – *New in Spikemark 3!*

Spikemark has two main modes of operation: **Simulator** and **Live**. The meaning of these modes is perhaps obvious, but the effect of switching from one mode to another is so profound that it warrants some discussion.

When the mode switch is set to **Simulator**, as shown in Figure 103, all actions are simulated in Spikemark. No motors will actually move, no network connections are made, and no cues will physically run. Instead, all motor movements are simulated mathematically in Spikemark and shown only on-screen so that you can visualize what a cue will look like, and how it will behave from the safety and comfort of a virtual stage. This allows you to try out new techniques, explore scenic motion with the other members of the design team, or present an idea to a director without having to setup literally tons of gear and clear a spot on the stage and time on your production calendar.

By contrast, when the mode switch is set to **Live**, as shown in Figure 104, all commands from Spikemark are sent over the Ethernet network to real **Stagehand** controllers which will move real scenery. In **Live** mode, every action on screen has very real, and potentially dangerous, consequences on the stage where real people are moving around.

The beauty of the **Simulator** is its simplicity and transparency. It works just like Spikemark, almost everything that you can do in **Live** mode works in **Simulator**. You can jog machines around the virtual stage, run cues, restore cues, adjust timings, etc. You can easily flip between **Simulator** and **Live** modes at the push of a button, allowing you to tweak the timing of a cue at the tech table in **Simulator** and then flip over to **Live** mode and see the result full-scale. It is a marvelous way to work, but it would be tragic to get confused about the mode in which you are operating and unexpectedly cause a collision on stage. To help prevent such an accident, here are a few tips:

1. The **Show Mode** switch is always visible in the upper right corner of the main window. Glance up there if you are ever unsure Spikemark's operating mode (Figure 102).

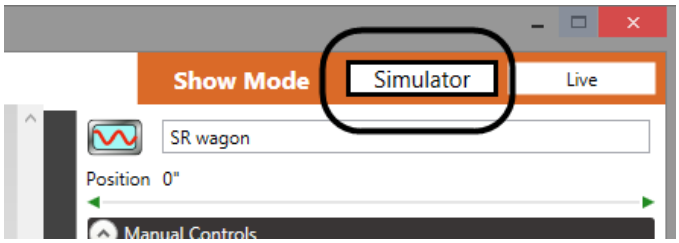


Figure 102

2. The background color of the **Stage Model Viewer** is light when Spikemark is in **Simulator** (Figure 103), and dark when in **Live** mode (Figure 104).

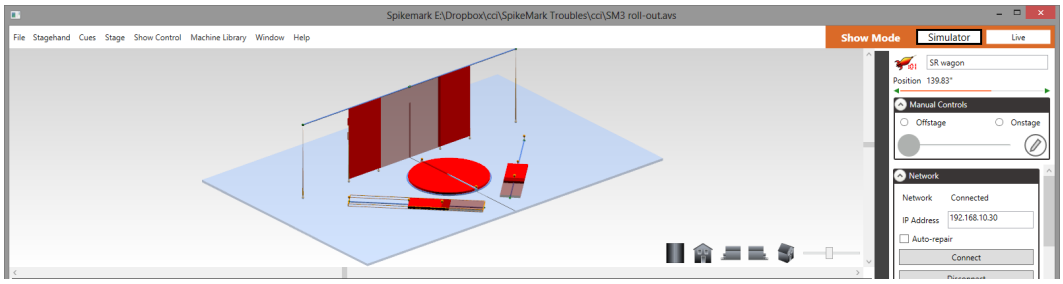


Figure 103

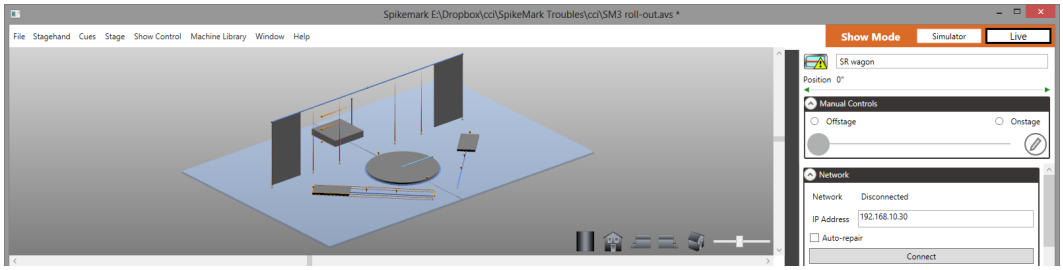


Figure 104

3. When you switch into **Live** mode, Spikemark will be disconnected from all machines on the network. This is done on purpose. You should actively connect to each machine on the network that you want to start using in **Live** mode. Having to take action to connect to the physical machinery on stage before running a cue should help remind you that you are about to move real, heavy scenery around real, fragile beings. Stay alert, be safe.

NOTE: To use Live Mode and move real scenery with real machinery, you will need to purchase a license. Simulator is free to use anytime. You may create shows in Simulator, save them, and share them. Have fun experimenting in Simulator.

For the rest of the manual, we won't distinguish between Simulator and Live mode. You can follow along in whichever mode makes sense for your circumstance, though it's often nice to try things out in **Simulator** first and later in **Live** mode.

The Stagehand Motor

Adding a New Stagehand Motor

Right-click in the **Cue Grid** and select **Add Motor** (Figure 105)**.**

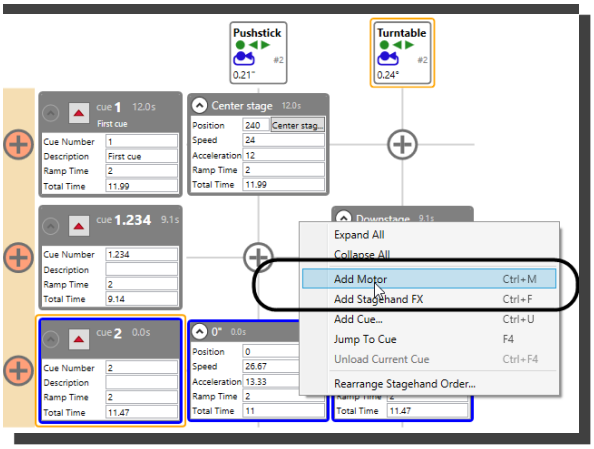


Figure 105

Or in the **Stagehand** menu, click **Add Motor** (Figure 106).

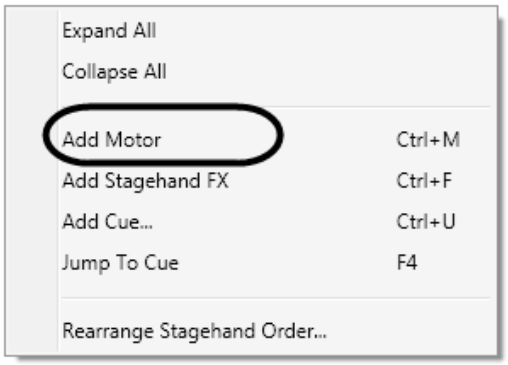


Figure 106

Or, press **Ctrl** + **M**.

NOTE: Spikemark 3 now has a Machine Library making it easy to add pre-configured machines to your show. See The Machine Library – *New in Spikemark 3!* starting on page 93 for more details.

Stagehand Motor Configuration

Each Stagehand Motor in a show has a collection of parameters that are typically set *once* during a production. Parameters include properties such as the name of the motor, the IP address of the motor, the motor's maximum speed, etc. The **Stagehand Editing Pane** on the right side of the screen allows you to edit all of the configuration parameters for any motor in the show.

To edit a Stagehand Motor's configuration parameters, select the motor in the **Cue Grid** by clicking on it. The selected Stagehand Motor is then highlighted in orange (Figure 107).

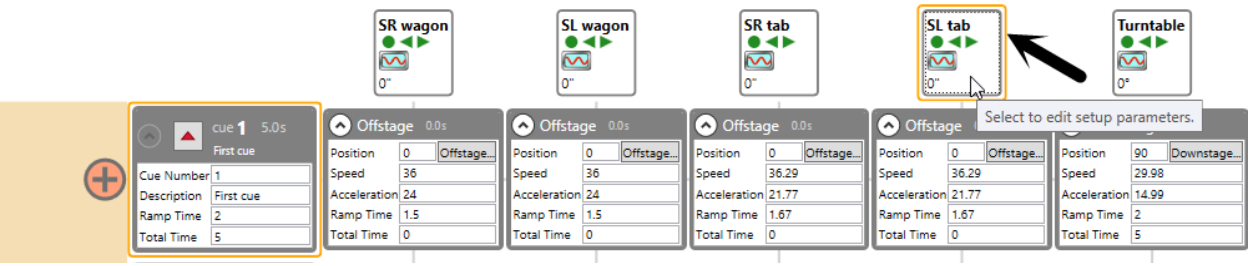


Figure 107

Once you have selected a motor in the **Cue Grid**, you can begin editing its parameters in the **Stagehand Editing Pane**. The **Stagehand Editing Pane** is a scrolling window that contains collapsible windows with each section grouped by function (such as Network, Position, Spikes, etc.). You can collapse or expand each section by clicking the up/down carrot top arrow in order to show or hide each section's detailed information (Figure 108).

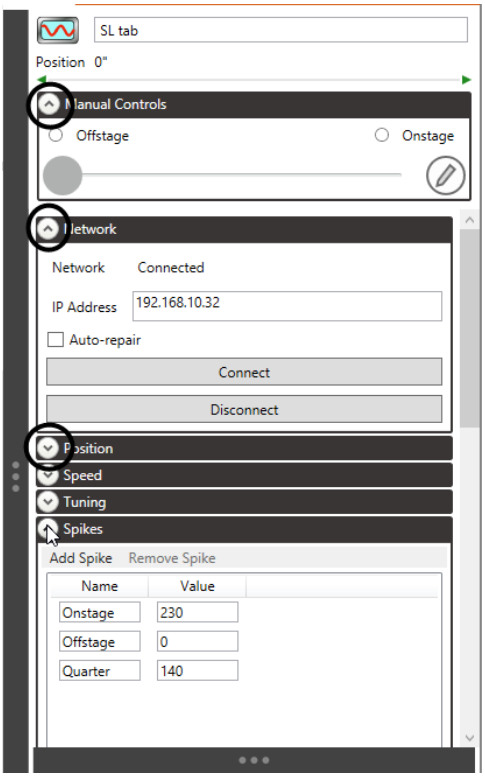


Figure 108

Renaming a Motor

To help you identify all the motors in your show, you can give each motor a name. Typically, motor names reflect the *purpose* of the motor and makes writing cues more intuitive.

1. Select the motor to be edited in the **Cue Grid** and then click into the field in the **Stagehand Editing Pane** that contains the default motor name.
   In the example below, the default name is "Motor 1" (Figure 109).

Figure 109

2. Type a new name.
   In the example below, the new name is "SL Wagon" (Figure 110).

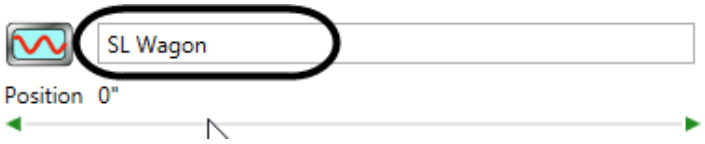


Figure 110

3. Press the **Tab** key or click out of the field, to update the name. The new motor name will be updated in the **Cue Grid** (Figure 111).

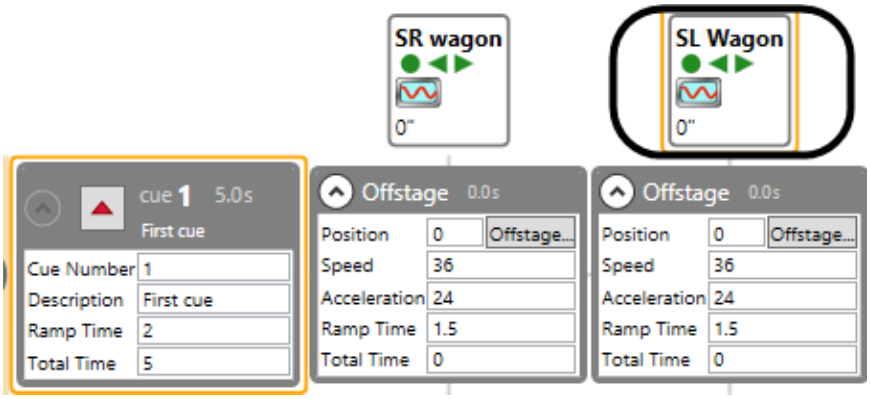


Figure 111

Motor Status

Motor status information is displayed next to the motor name in the **Stagehand Editing Pane** and in the **Cue Grid** column headers. Each icon status is described below.

Icon	Status
	Motor disconnected from the network.
	Motor connected to the network, but idle.
	Motor is inactive and won't run cues or affect cue links
	Motor has completed a cue.
	Emergency stop.
	Cue loaded.
	Soft stopping, but not yet fully stopped.
	Stopped mid-cue by a soft-stop command.
	In manual mode with the motor enabled, but zero-speed.
	Moving forward in manual mode.
	Moving reverse in manual mode.
	Running a cue forward in target-tolerance mode.
	Running a cue forward in strict-timing mode.
	Running a cue reverse in target-tolerance mode.
	Running a cue reverse in strict-timing mode.
	Stopped by a forward limit switch on the motor.
	Stopped by a reverse limit switch on the motor
	Stopped by a position error fault.

Manually Moving a Motor Using the On-Screen Controls

NOTE: You will frequently need to move a motor forward or backward without running a cue. During initial setup, tech, and work calls the scenery often needs to be nudged around the stage. You can either run the motor using the physical jog buttons on the Stagehand, or you can use Spikemark's on-screen Manual Controls **to move the motor, as described in the steps below.**

1. Select the motor you wish to move in the **Cue Grid**.
2. Confirm that the motor is connected to the network. If it's not connected, establish a network connection (see "Setting Up a Network Connection" on page 41).
3. In the **Stagehand Editing Pane** on the right side of the screen, locate the **Manual Controls** section (Figure 112).

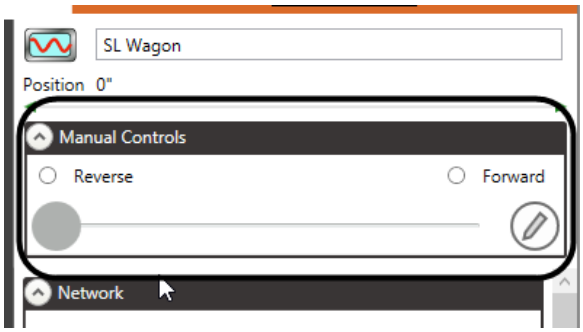


Figure 112

4. Select a direction, either **Forward** or **Reverse** (Figure 113).
   Note: You can customize the direction labels by pressing the pencil-icon button.

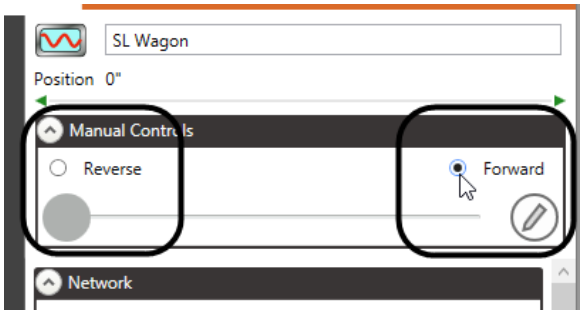


Figure 113

5. Click and drag the slider slowly to the right, which will release the brake and ramp up the motor speed (Figure 114).

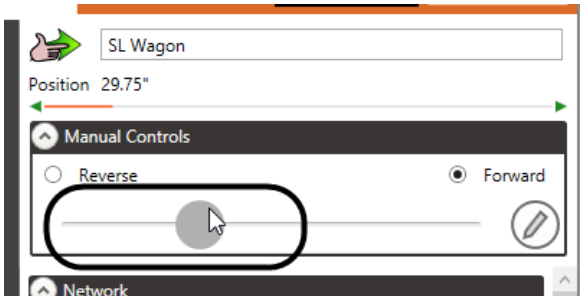


Figure 114

6. To stop the motor, slowly move the slider to the left and then release the mouse button. The motor will stop and the brake will engage.

NOTE #1: While manually jogging a motor in the Forward direction, the encoder counts should be increasing.

While manually jogging the motor in the Reverse direction, the encoder counts should be decreasing.

If this is NOT the case, you should correct the wiring by swapping the A encoder wire w/ the B encoder wire and swapping the /A encoder wire w/ the /B

NOTE #2: While manually jogging a motor forward, if the machinery is traveling in the wrong direction [for example, you want a piece moving forward ONSTAGE, but it's moving OFFSTAGE] and the encoder counts are decreasing, then you need to swap the two power wires so that it goes in the correct direction and the counts increase.

If NOTE #2 applies to a DC motor: switch the A^+ wire with the A^- wire.

If NOTE #2 applies to an AC motor: switch any two of the three power wires

Do NOT switch the ground wire!

Editing Motor Direction Labels – *New in Spikemark 3!*

Depending on the scenery you are moving, the Manual Control direction labels "Forward" and "Reverse" may not be descriptive. You can change the labels to read "Onstage" and "Offstage", or "Clockwise" and "Counter-clockwise", or "Up" and "Down".

1. Select the machine you wish to edit in the **Cue Grid** or **Stage Model Viewer**.
2. Click the **Edit** button in the **Manual Controls** section of the **Stagehand Editor Pane** (Figure 115).

Figure 115

3. Enter the new names for the direction labels (Figure 116).

Figure 116

4. Click the **Edit** button again to accept the changes (Figure 117).

Figure 117

Scaling Motor Position

Whenever a piece of scenery moves, an encoder connected to that scenery spins. As the encoder spins, it generates a stream of electrical pulses. The Stagehand connected to the encoder counts those pulses to determine where the scenery actually is. The Stagehand relays the position back to Spikemark, which then displays the position information on screen. By default, Spikemark displays the information as raw encoder counts. Viewing the position of the scenery as raw encoder counts is almost never ideal. Usually you'll want to see the motor's position displayed in something more meaningful, such as inches or degrees.

Motor scaling allows you to specify a ratio between raw encoder counts and physical units. With a position scale defined you can view the motor's current position and write cues in a more user-friendly manner. Spikemark can display position as inches, feet, millimeters, meters or degrees. Therefore, rather than having to type 1,245,120,295 encoder counts, you would simply enter 123.5 inches or 360 degrees.

Setting Position Scale for a Winch

1. Confirm that the current **Position Scale** field is set to “1” in the **Position** section of the **Stagehand Editing Pane** (Figure 118).

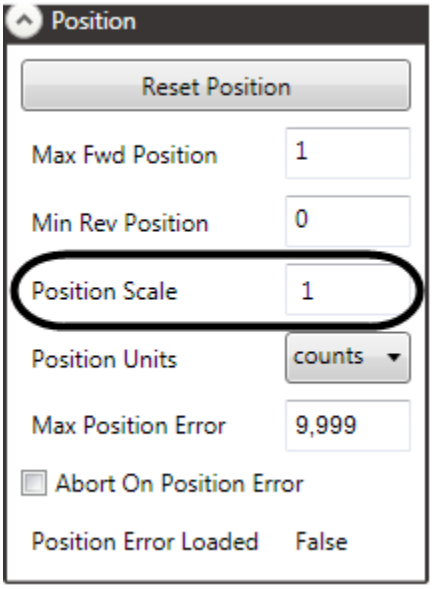


Figure 118

2. Click the **Reset Position** button (Figure 119).

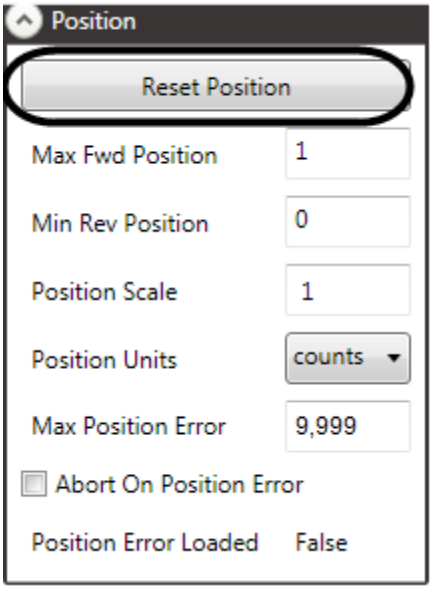


Figure 119

3. Enter “0” in the **New Position** field and click **OK** (Figure 120).

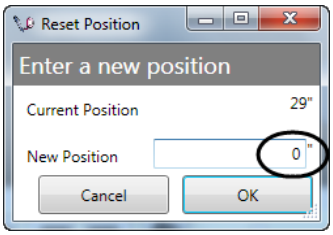


Figure 120

4. In the **Manual Controls** section of the **Stagehand Editing Pane** (Figure 121) run the motor forward or reverse as far as you safely can.

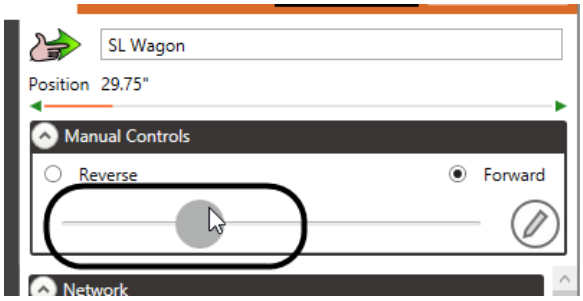


Figure 121

5. At the top of the **Stagehand Editing Pane**, note the number of encoder counts that are traveled.

6. Measure the physical distance traveled by the scenery.

7. To compute the scaling factor, divide the number of encoder counts traveled by the physical distance.
For example: 1,245,012 encoder counts / 125.5 inches = 9,920.41 encoder counts per inch.

8. Enter the result from the previous step into the **Position Scale** field and then select the appropriate units from the **Position Units** drop-down menu (Figure 122).
In this example, you would enter "9,920.41" and select "inches."

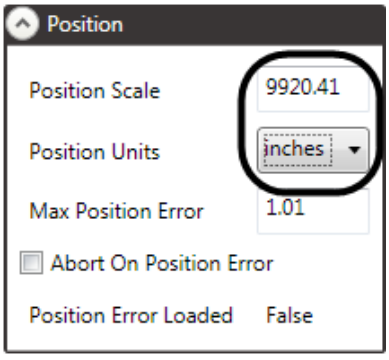


Figure 122

Setting Position Scale for a Turntable

1. Confirm that the current **Position Scale** field is set to “1” in the **Position** section of the **Stagehand Editing Pane** (Figure 123).

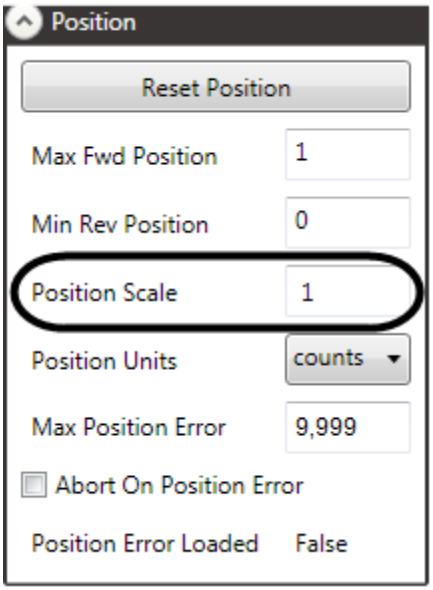


Figure 123

2. Click the **Reset Position** button (Figure 124).

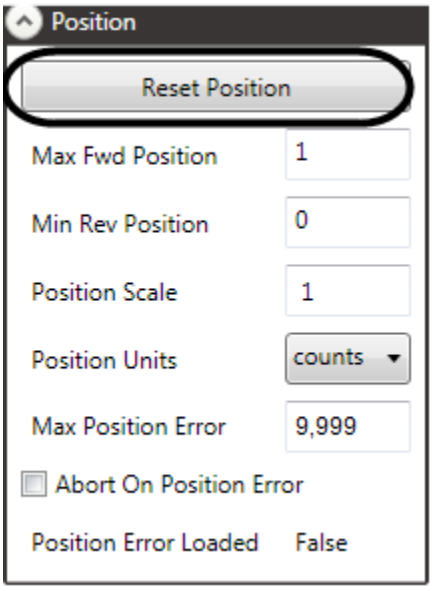


Figure 124

3. Enter “0” in the **New Position** field and click **OK** (Figure 125).

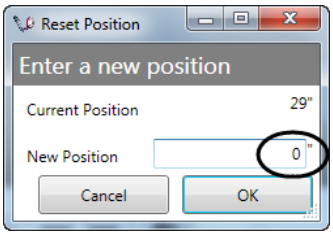


Figure 125

4. Place a piece of spike tape on the turntable deck and a matching piece of spike tape on the stage floor so you can register the turntable's "0" position.

5. Assuming you can run the turntable indefinitely in either direction, use the Manual Controls in the Stagehand Editing Pane to run the turntable at four (4) complete revolutions in the Clockwise (or Forward) direction.

6. Slow down and stop the turntable so that the spike tape marks you made in Step 4 line up.

7. Compute your position scale by dividing the number of encoder counts traveled by 1440 degrees. In our example:

$$2{,}408{,}697 \text{ counts} / 1440 \text{ degrees} = 1672.94016 \text{ counts/degree}$$

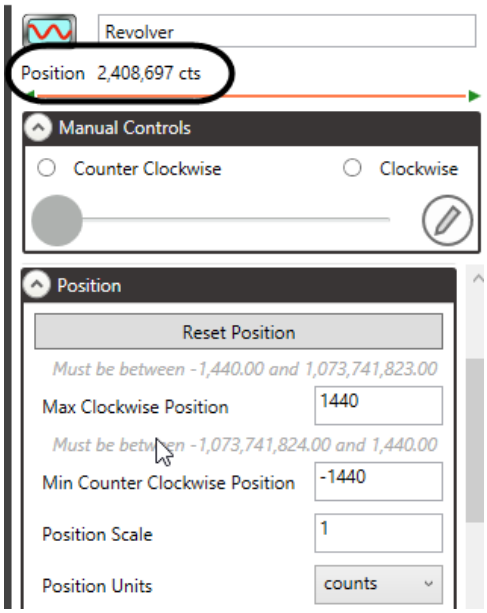


Figure 126

8. Enter the **Position Scale** and change the **Position Units** to **degrees**. Notice that the **Position** value updates to 1440 degrees (Figure 127).

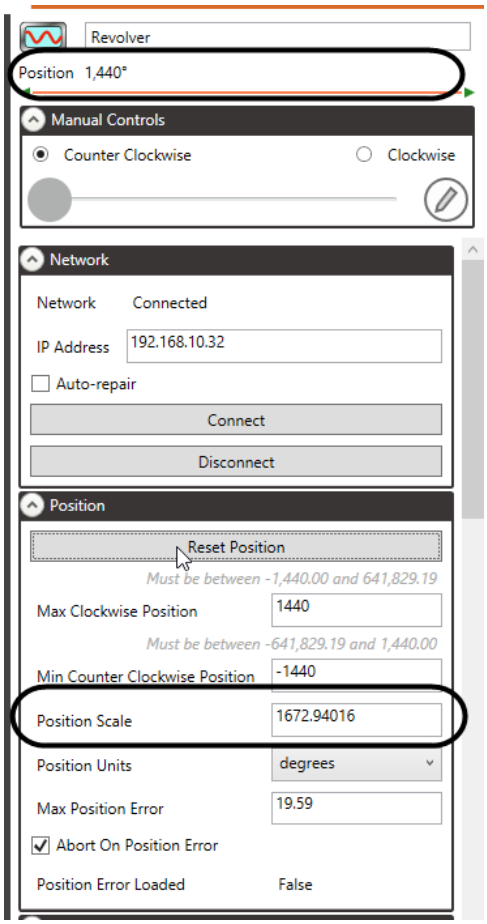


Figure 127

Setting Motor Software Limits: The Min Reverse and Max Forward Positions

All motors in Spikemark *must* have a **Max Forward Position** and a **Min Reverse Position** defined in the **Position** section of the **Stagehand Editing Pane.**

> NOTE: **Spikemark calculates the track length for all winches in the Stage Model Viewer based on these configuration parameters. By setting these values, Spikemark will prevent you from writing cues that travel beyond the physical limits of the machine. If you enter in a position of 1000" for a cue, but the** Max Forward Position **is set to 100", Spikemark will reduce the cue target position to 100" to keep it within the defined limits.**

1. Select the motor in the **Cue Grid**.

2. To set the **Min Reverse Position**, go to the **Manual Controls** section in the **Stagehand Editing Pane** and run the motor in reverse as far as its physical limit will

allow. *Observe the **Position** value at the top of the **Stagehand Editing Pane** (circled in Figure 128).*

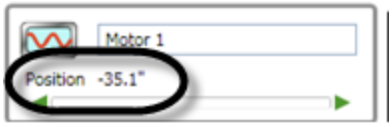


Figure 128

3. Type the **Position** value into **Min Reverse Position** field (Figure 129).

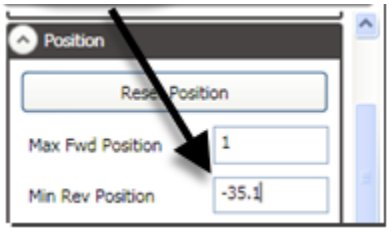


Figure 129

4. To set the **Max Fwd Position**, go to the **Manual Controls** section of the **Stagehand Editing Pane** and run the motor forward as far as its physical limit will allow. *Observe the **Position** value at the top of the **Stagehand Editing Pane** (circled in Figure 130).*

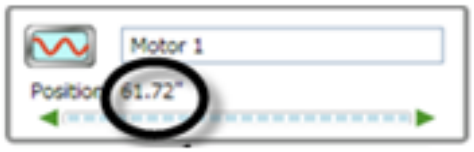


Figure 130

5. Type the **Position** value into **Max Fwd Position** field (Figure 131).

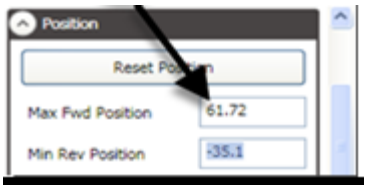


Figure 131

NOTE: The following rules must be followed for Max Forward Position **and** Min Reverse Position:

The Max Forward Position must be greater than Min Reverse Position**.**

Values can be negative, provided that the above rule is followed.

Turntables must have software limits, even if there is no physical limitation to how far they can travel. For turntables, you can often just set a high value in both directions [For example: 7200 degrees for the Max **Forward Position and -7200 degrees for** Min Reverse Position **would allow the turntable to spin around in either direction 20 times].**

If possible, set the Max Forward Position and the Min Reverse Position to be slightly less than the physical limits set on your motor switches (Figure 132).

Physical limits *usually* do not apply to turntables.

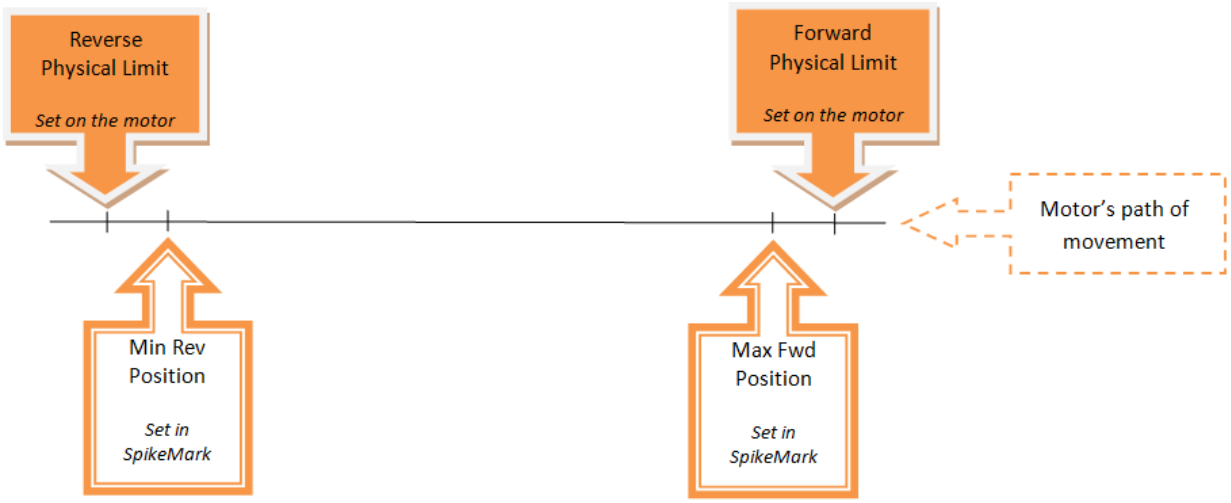


Figure 132

Resetting the Motor's Position Value

Motor position is stored in the Stagehand. Spikemark simply displays the position information that it receives from the Stagehand. There are times when you need to change the position information that is stored in the Stagehand. For instance, if the Stagehand loses power or is replaced then the position information will be wrong. In such a case, you should move the motor to a known position with the **Manual Controls** on the stage and then reset the position value stored inside the Stagehand.

NOTE: If your production powers down every evening, you will need to reset every motor's position value before every show.

1. Select the Motor in the **Cue Grid** and confirm that it is connected to the network (see "Setting Up a Network Connection," page 41).

2. Click the **Reset Position** button in the **Position** section of the **Stagehand Editing Pane** (Figure 133).

Figure 133

3. In the **New Position** field, enter the *correct* **Position** value and click **OK** (Figure 134).

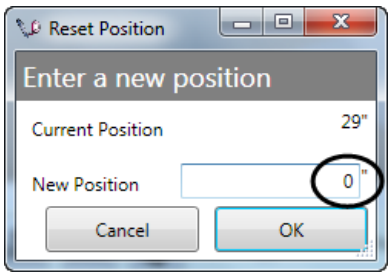


Figure 134

Motor Tuning

Motor tuning is perhaps the most difficult aspect of scenic automation. By tuning a motor, you are describing to a Stagehand how it should regulate the motor speed during a cue using mathematical formulae. We won't get too deep into the intricacies of that math, but it is hard to discuss motor tuning without getting a little involved in the nuts and bolts of motion control theory.

First, it helps to understand the problem that motor tuning solves. The best analogy for motor tuning is cruise control on a car. When you set your cruise control to 55 mph, the car has to adjust the amount of fuel going to the engine to keep moving at that speed. If you are going up hill, the car lags for a second and then more gas rushes into the engine and the car speeds up to get back to the desired speed.

Motion control for scenery is very similar. We program the Stagehands with a speed and target position, and the Stagehand has to adjust power to the motor to achieve the desired effect. If there's a bump in the floor, the winch has to have a little more power to maintain speed. If the winch has too much power, it will lurch over the bump and then quickly reduce power so that it doesn't go too fast once it is over the bump.

The Stagehand is constantly analyzing where the motor is versus where it should be and then adjusting motor power to minimize the difference between where the motor should be and

where it really is. It does this analysis a few million times per second. When it wants to apply power to correct for error in position, it looks to us for guidance.

By entering in some tuning parameters, we are giving the Stagehand that guidance. In a confounding abstract way, we are specifying how much power to give the motor when it needs to make a correction. If the values that we enter give the motor too much power during correction, the motor will be jerky as it over-corrects and then has to pull back (remember, this happens millions of times per second). If the values we enter do not provide enough power to the motor to correct position, it will never reach the cue position since it will run out of power and be unable to muscle the load onto the target.

There are five tuning values that can be used to achieve smooth, accurate motion. To adjust these parameters, select the motor you wish to tune in the **Cue Grid** and select the **Tuning** section of in the **Stagehand Editing Pane** (Figure 135 through Figure 140).

Proportional Gain

Proportional Gain (Figure 135) is the ratio of power to position error. The higher this value, the more power will be applied to correct for position errors. This is often the only tuning parameter you will need to adjust. **Proportional Gain** needs to be at least "1" for the motor to move at all. If the value is too high, the motor will begin lurching. Sometimes this lurching can be extreme so make sure to have your hand on the Emergency Stop when you first adjust this parameter.

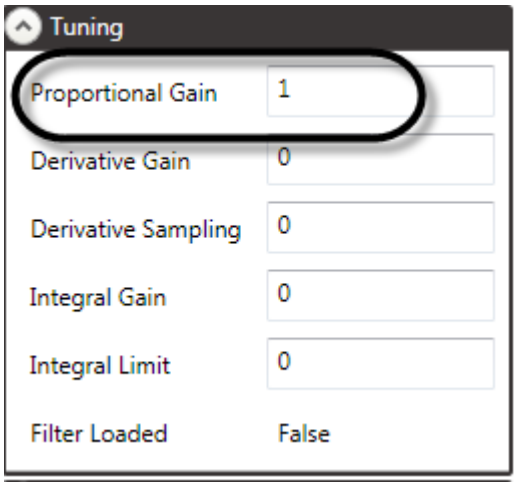


Figure 135

Derivative Gain

Derivative Gain (Figure 136) is applied against changes in position error. It can have a dampening effect on a motor, absorbing the shocks from a high **Proportional Gain.**

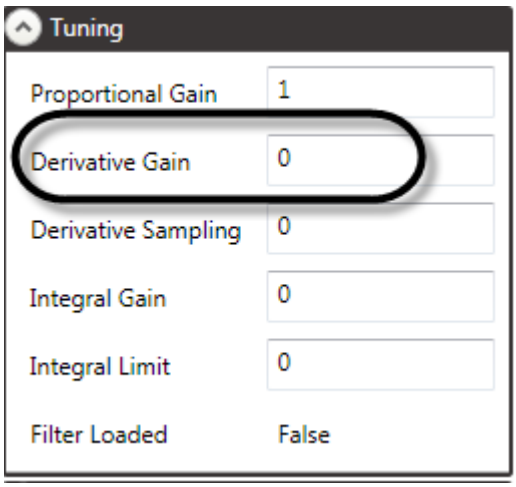


Figure 136

Derivative Sampling

Derivative Sampling (Figure 137) describes how often the **Derivative Gain** should be applied. If you are using **Derivative Gain**, **Derivative Sampling** must be set to at least "1." Rarely is there any benefit to using a **Derivative Sampling** value higher than "2."

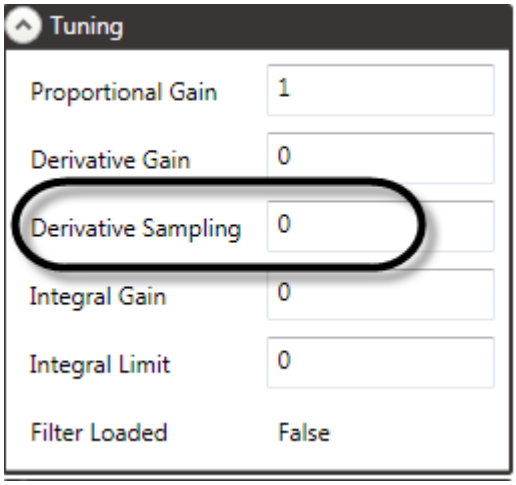


Figure 137

Integral Gain

Integral Gain (Figure 138) tries to compute corrective action by evaluating the position error and motor response over time. From a practical standpoint, **Integral Gain** will often correct the motor position at the end of travel. This can be helpful if you need to reduce **Proportional Gain** to smooth out a motion, but then loose accuracy. In such a scenario, adding in 1 or 2 points of **Integral Gain** will often retain accuracy without sacrificing smooth travel.

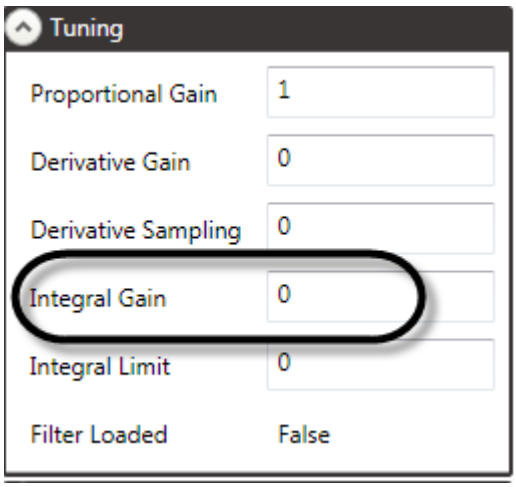


Figure 138

Integral Limit

Integral Limit (Figure 139) prevents a condition known as Integral Windup, which can lead to really erratic motor behavior. If you use an **Integral Gain**, set **Integral Limit** to 1000.

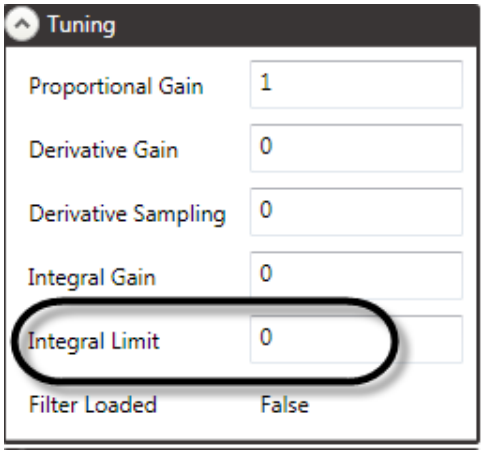


Figure 139

Filter Loaded

The **Filter Loaded** field (Figure 140) is *not* an adjustable motor tuning parameter. Rather, it indicates whether the filter parameters have been sent to the motor yet. After you change a tuning parameter, that change is saved until the next cue is run. When the next cue is run, the new filter parameters will be sent to the motor automatically.

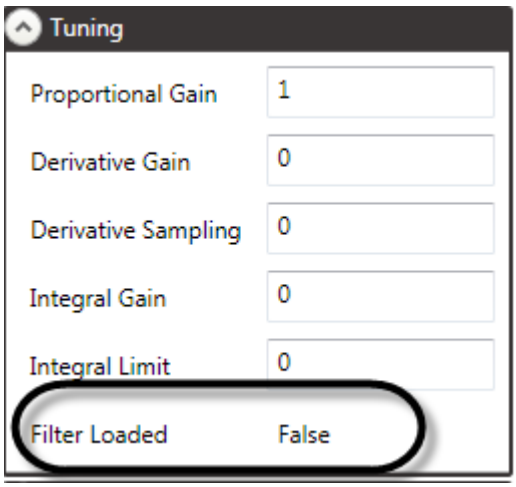


Figure 140

Over the past several years, we've come up with a practical approach to motor tuning that yields good results in a reasonable amount of time. That approach is shown in the form of a flow chart (Figure 141). The steps are clear and simple, but it can still be a frustrating experience the first time you attempt to tune a motor. It's extremely important to find at least an hour of quiet time on the stage without anyone else working around the moving scenery. Get a cup of coffee, relax, focus and take a stab at it. If that doesn't work, give us a call (see the "Contact Us" section on page 207). We have had plenty of experience helping people tune motors over the phone. The good news is that it gets a lot easier once you see it work the first time.

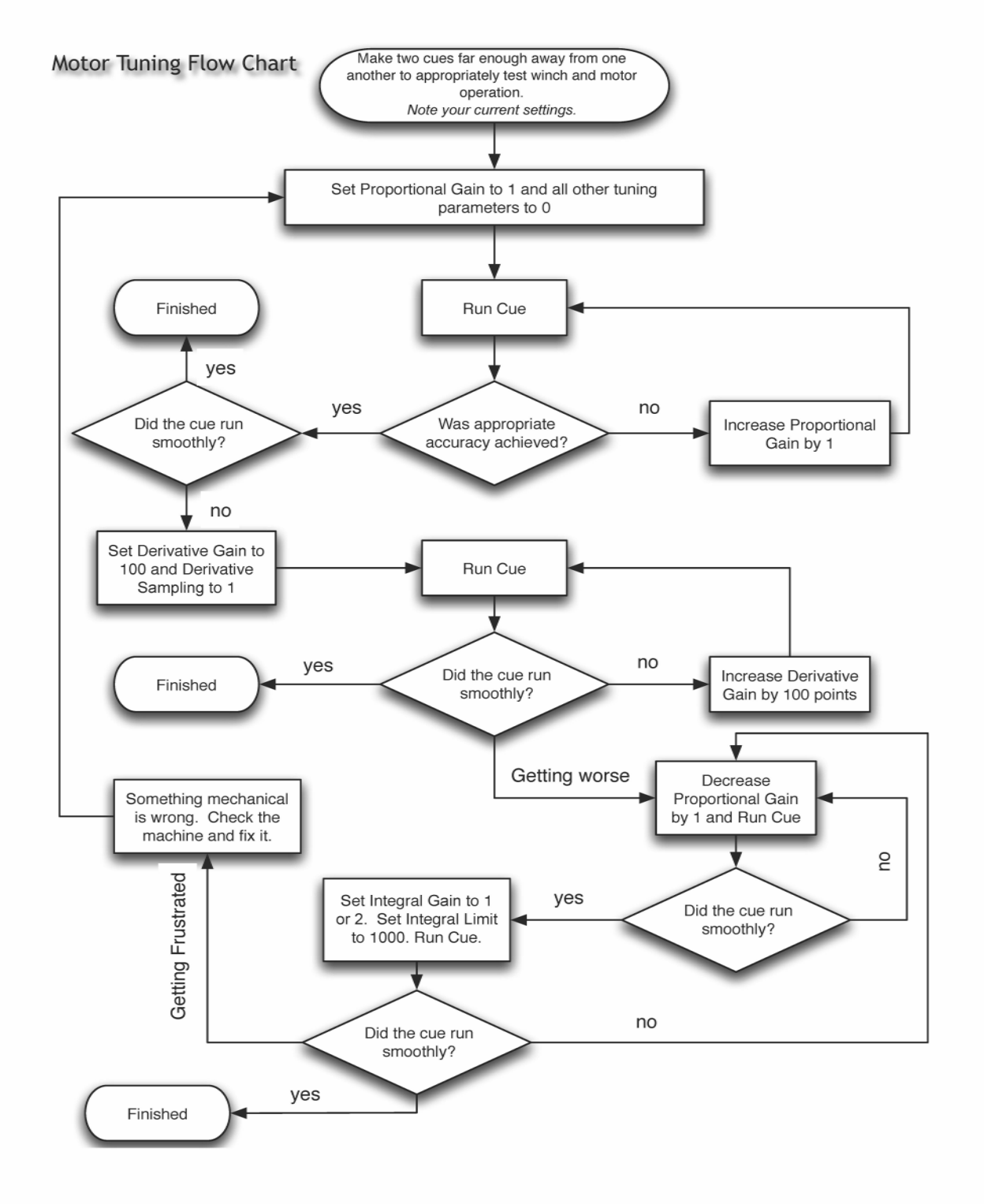

Motor Tuning Flow Chart
Make two cues far enough away from one another to appropriately test winch and motor operation.
Note your current settings.
Set Proportional Gain to 1 and all other tuning parameters to 0
Run Cue
Was appropriate accuracy achieved?
yes
no
Increase Proportional Gain by 1
Did the cue run smoothly?
yes
Finished
no
Set Derivative Gain to 100 and Derivative Sampling to 1
Run Cue
Did the cue run smoothly?
yes
Finished
no
Increase Derivative Gain by 100 points
Getting worse
Decrease Proportional Gain by 1 and Run Cue
Did the cue run smoothly?
yes
Set Integral Gain to 1 or 2. Set Integral Limit to 1000. Run Cue.
no
Did the cue run smoothly?
yes
Finished
no
Getting Frustrated
Something mechanical is wrong. Check the machine and fix it.

Figure 141

Setting the Max Position Error

Once a motor has been adequately tuned, it is a good idea to set a **Max Position Error** value and make sure the **Abort On Position Error** field is selected in Spikemark. This protects you from runaway motors when the encoder is damaged or disconnected.

Without a **Max Position Error**, the Stagehand will send an increasing amount of power to compensate for a position error. If the encoder is not sending information to the Stagehand, the Stagehand will think that the motor is not moving. So it will rapidly work up to full speed and run at full speed indefinitely, since, from the Stagehand's perspective, the motor is standing still. Without setting **Max Position Error,** you will be forced to use the Emergency Stop (but not before you have been scared out of your skin).

A **Max Position Error** value can also stop a motor when it is poorly tuned, but not posing any danger. This interrupts the show and can be detrimental to the performance. To keep the motor from faulting during normal circumstances, the default value for maximum position error is set as high as possible. You can tighten the allowable position error by lowering this value if you have requirements for a more exacting motion profile.

NOTE: The maximum allowable Max Position Error [MPE] value is 32,767 raw encoder counts after scaling.

The MPE is given in scaled position units such as inches or degrees

Therefore, if you have a position scale of 10,000 encoder counts per inch, your MPE will equal 3.2767" [32,767 raw encoder counts ÷ 10,000 encoder counts/inch = 3.2767 inch].

The MPE of 32,767 requires a Stagehand running firmware v1.7 or higher. Prior to firmware v1.7, Stagehands were limited to an MPE value of 9,999.

1. Select the motor you wish to adjust in the **Cue Grid**.
2. Check the **Abort On Position Error** option in the **Position** section of the **Stagehand Editing Pane** (Figure 142). If you like, you can tweak the value of the **Max Position Error** (this value is given in scaled position units such as inches or degrees, etc.).

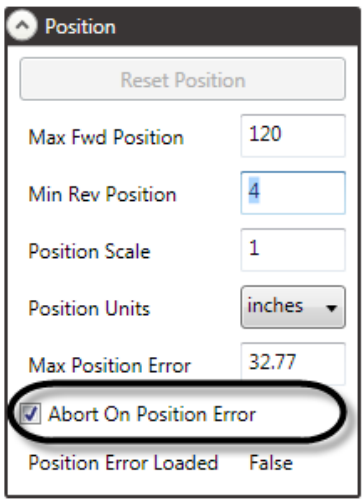


Figure 142

NOTE: Position error changes are loaded into the Stagehand prior to running a cue.

Setting the Motor Cue Completion Mode

Spikemark has two modes for running motors:

Target Tolerance

Target Tolerance mode makes motor cues more reliable regardless of poor tuning. When using **Target Tolerance** mode, the motor will continue to try to achieve target regardless of how much time has passed. Once the motor position is within your specified tolerance, it will turn off and display a **Complete** status icon.

NOTE: Target Tolerance is the default mode in Spikemark.

Strict Timing

Strict Timing mode insists that the motor move accurately and achieve the cue position within the theoretical time allowed for that movement based on the programmed distance, speed, and acceleration. If the motor cannot achieve target in the computed time, it will be turned off regardless of its current position.

NOTE: For shows that have tight timing requirements, this option may be useful. However, it has mostly been included in Spikemark since it was the only mode available in earlier versions of Spikemark. When using this mode, you have to be diligent about tuning your motors well. Sloppy tuning will result in motors abruptly stopping in mid-move.

1. To specify which cue completion mode to use, select the motor you wish to edit in the **Cue Grid**.

2. Scroll to the **Cue Behavior** section in the **Stagehand Editing Pane** (Figure 143).

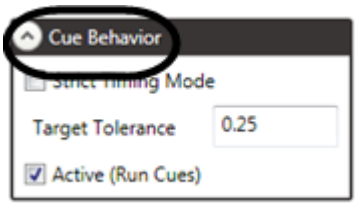


Figure 143

3. To use **Target Tolerance** mode, make sure that **Strict Timing Mode** is *deselected* and enter a **Target Tolerance** value**,** which is listed in scaled position units (for example, inches, degrees, etc.).
In Figure 144, "0.25" equals 1/4 inch.

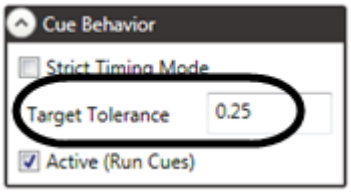


Figure 144

4. To use **Strict Timing Mode,** select the **Strict Timing Mode** field (Figure 145).
*With **Strict Timing Mode** checked, the **Target Tolerance** value will be ignored.*

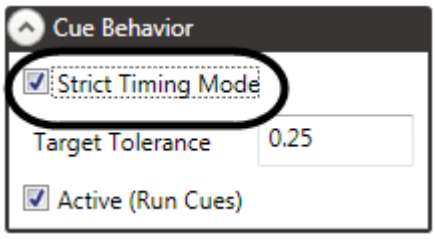


Figure 145

Setting a Motor's Active Mode

Most of the time, you will want the motor to load and run cues normally. However, there are occasions when a motor needs to be temporarily removed from the show. Perhaps there is a problem with the scenery on a winch track, or perhaps you need to run through movements using only the motors downstage and don't want any upstage motors to move.

To temporarily disable a motor, deselect the **Active** box (Figure 146) in the **Cue Behavior** section of the **Stagehand Editing Pane**. The motor will not run cues until the **Active** box is checked again.

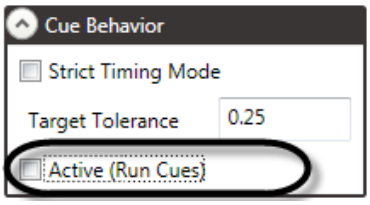


Figure 146

You can also set the Active mode by right-clicking on the Cue Grid header as shown in Figure 147.

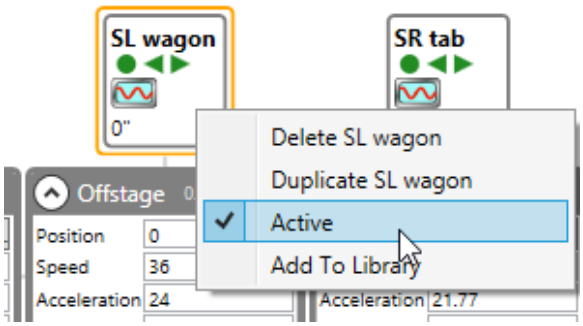


Figure 147

When a motor is deactivated (not **Active**), it will not impede the operation of cue *links* (see "Cue Links" on page 134). Normally a cue needs to successfully load all motor movements before it can be executed by a cue link. However, if a motor within in a cue is *not* **Active** and thereby unable to move, the entire cue will *still* be allowed to execute. Spikemark will trust that *you have ascertained that it is safe for the cue link to run without this motor* and you are overriding the normal requirement that all motors must be able to move before any motor will move in a linked-cue sequence.

EXAMPLE:

1: Assume you have written Cue 1, in which WAGON A moves upstage **and** a CURTAIN goes up.

2: Also assume you have written Cue 2, in which WAGON B moves downstage, past the CURTAIN and meets WAGON A.

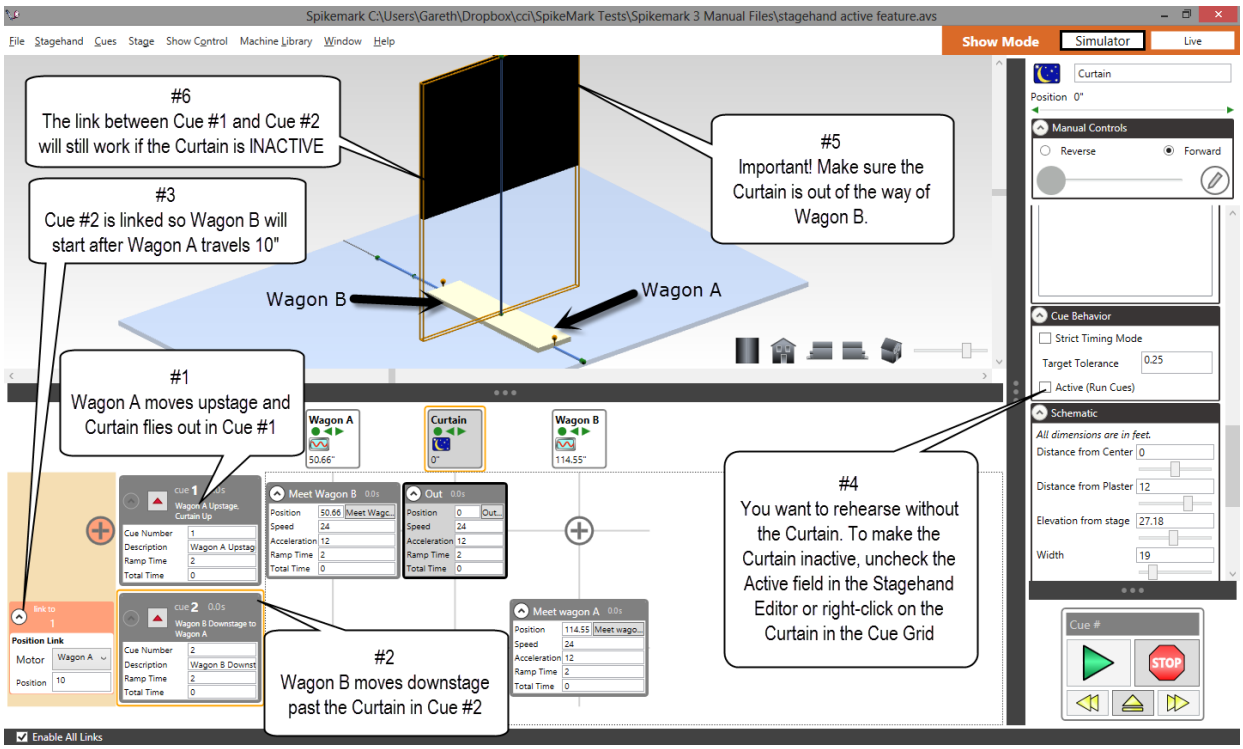
3: Now assume that Cue 1 and Cue 2 are linked—by position—so that WAGON B starts moving once WAGON A has moved [for this example, 10”].

4: Lastly, assume that you want to rehearse WITHOUT the CURTAIN.

5: To do this, you would fly the curtain up and make sure it is out of the way of WAGON B. Next, you would deactivate the CURTAIN [by deselecting the **Active** field in the **Cue Behavior** section of the curtain’s **Stagehand Editing Pane**]. Then it would then be safe to test WAGON A and WAGON B, as you’ve moved the curtain up and out of the way of WAGON B

6: Deactivating a motor does NOT stop the links between cues. Therefore, the link between Cue 1 and Cue 2 will still work [in this example, WAGON B will start moving once WAGON A has reached the 10” mark].

Also see *Figure 148*.


#6
The link between Cue #1 and Cue #2 will still work if the Curtain is INACTIVE
#3
Cue #2 is linked so Wagon B will start after Wagon A travels 10"
#5
Important! Make sure the Curtain is out of the way of Wagon B.
Wagon B
Wagon A
#1
Wagon A moves upstage and Curtain flies out in Cue #1
#4
You want to rehearse without the Curtain. To make the Curtain inactive, uncheck the Active field in the Stagehand Editor or right-click on the Curtain in the Cue Grid
#2
Wagon B moves downstage past the Curtain in Cue #2
Show Mode
Simulator
Live
Manual Controls
Cue Behavior
Schematic
Enable All Links
Figure 148

It is worth mentioning here that the **Active** state of a motor can also be controlled through a Stagehand FX input action. Using an input on a Stagehand FX, you can quickly deactivate one or many motors. (To learn more about Stagehand FX input actions, see *Deactivating Stagehand FX Action* section on page 93).

Changing the Status Refresh Rate

Whenever Spikemark connects to a motor, it starts asking the Stagehand for updated information about its status, position, limits, and Emergency Stop condition. Spikemark asks for updated information several times a second to give the illusion of a steady stream of information. This is how the position information appears to fluidly change as the motor moves on stage. Most of the time, this all works great and you don't have to think about *how* it works. However, if you have a large show, or a slow computer, you may find that Spikemark seems to be slowing down or feeling clunky.

In the case described above, you may wish to tweak the frequency of information updates, known as the **Polling Interval,** so that Spikemark doesn't get bogged down with this constant updating process.

1. Select the motor you wish to adjust in the **Cue Grid** or **Stage Model**.
2. Scroll to the **Advanced** section of the **Stagehand Editing Pane** (Figure 149).

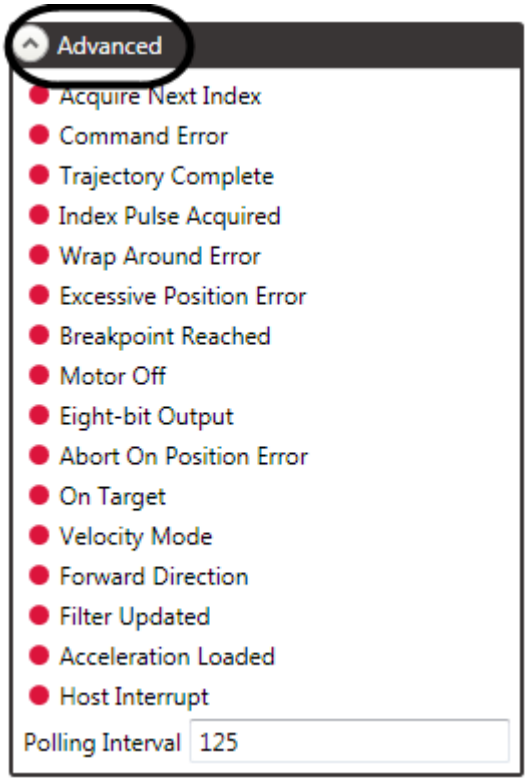


Figure 149

3. Enter a new value in the **Polling Interval** field (Figure 150).
   The values are in milliseconds. The default value is "125," which means that Spikemark updates motor information 8 times a second. Setting the Polling Interval to "250" will update motor information 4 times a second, which will relieve the computer of some processing stress, but the updates on screen will appear jerky. Setting the polling interval to 50 will provide updates 20 per second making the information updates very smooth and snappy, but eating more of your computer's resources.

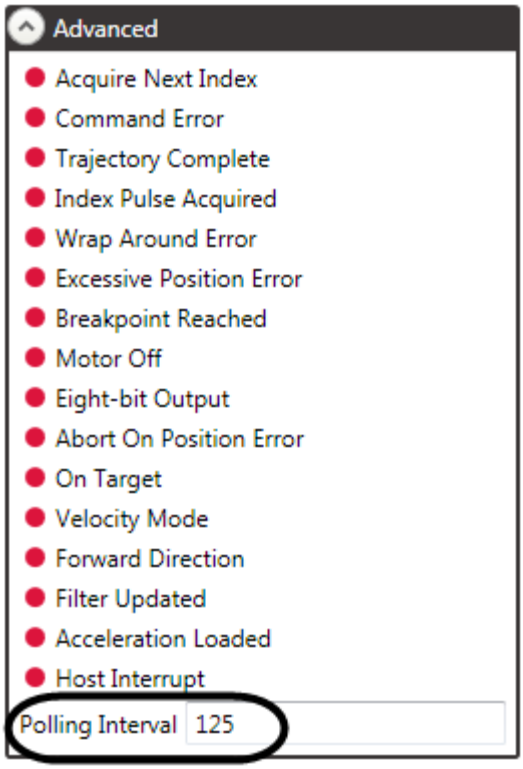

Advanced
Acquire Next Index
Command Error
Trajectory Complete
Index Pulse Acquired
Wrap Around Error
Excessive Position Error
Breakpoint Reached
Motor Off
Eight-bit Output
Abort On Position Error
On Target
Velocity Mode
Forward Direction
Filter Updated
Acceleration Loaded
Host Interrupt
Polling Interval
125

Figure 150

The Machine Library – *New in Spikemark 3!*

The preceding section detailed all of the various configuration settings that can be made for each machine that you wish to control with Spikemark. No doubt you get a sense that the initial setup and configuration of a machine can be time-consuming. The **Machine Library** allows you to store and retrieve machines with their parameter configuration so that you don't have to re-enter all of the settings each time you want to use a machine. Spikemark comes loaded with all of Creative Conners's machines: **Pushstick**, **Revolver**, and **Curtain Call** so if you are using one of our machines you can jump into writing cues immediately. If you are using your own machine connected to a **Stagehand AC**, **Stagehand DC**, or **Stagehand Mini**, you can still take advantage of the **Machine Library** by setting all the configuration parameters as outlined in the previous section and then saving your own machine definition in the **Machine Library**. We'll take a look at using the default machines and adding your own machines in this section.

Adding a Pushstick Winch to the Show

1. Click or tap the **Machine Library** menu (Figure 151).

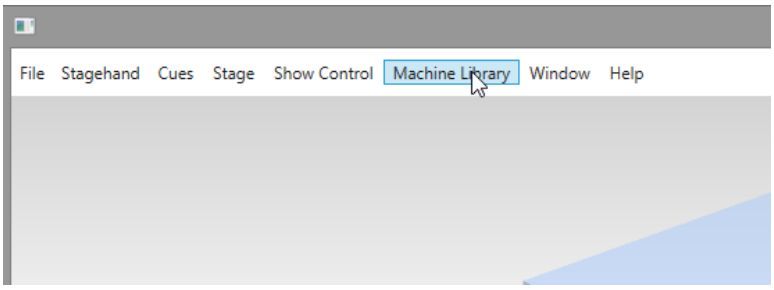


Figure 151

2. The **Machine Library** panel slides down from the top of the main window (Figure 152).

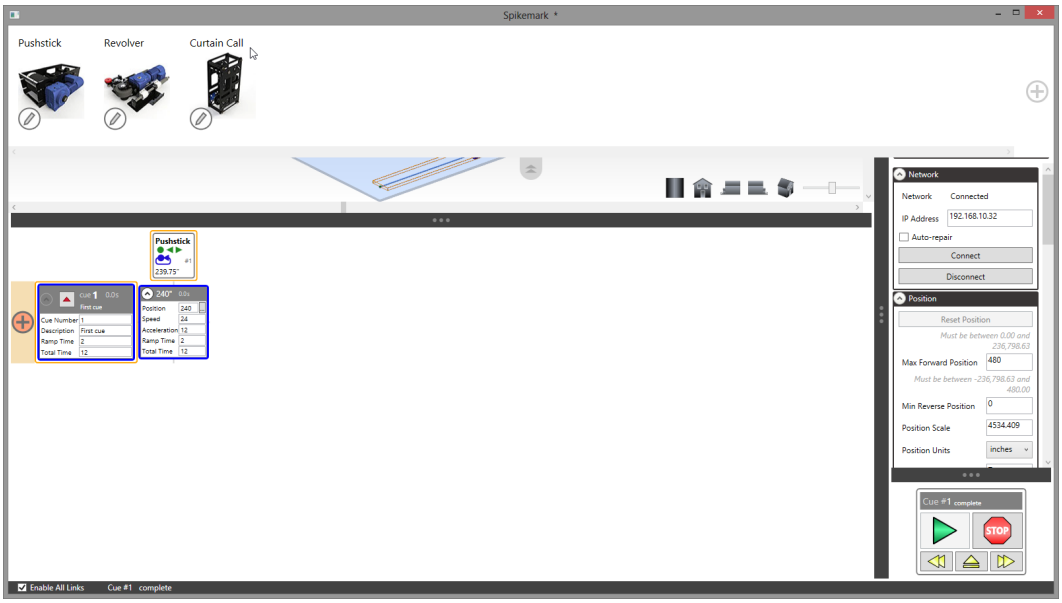


Figure 152

3. Drag the **Pushstick** machine from the **Machine Library** and drop it on either on the **Stage Model Viewer** or the **Cue Grid** (Figure 153).

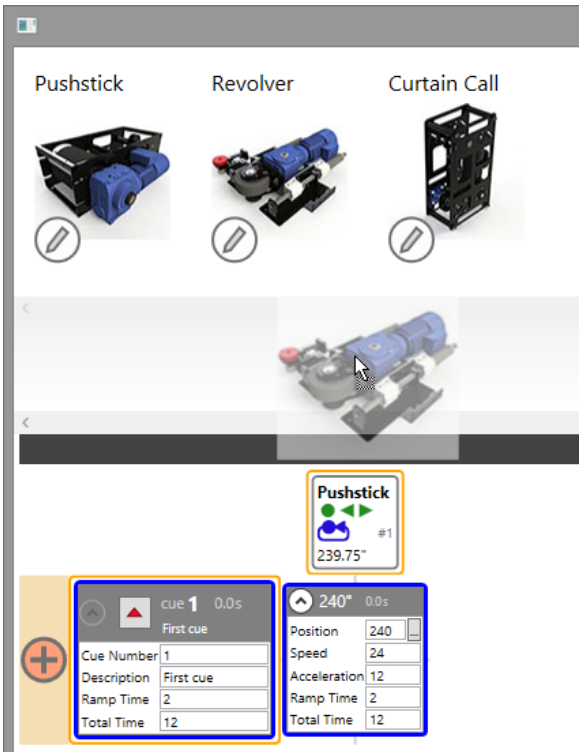


Figure 153

4. To close the **Machine Library** panel, either press the **Esc** button or click the **Machine Library** close tab (Figure 154).

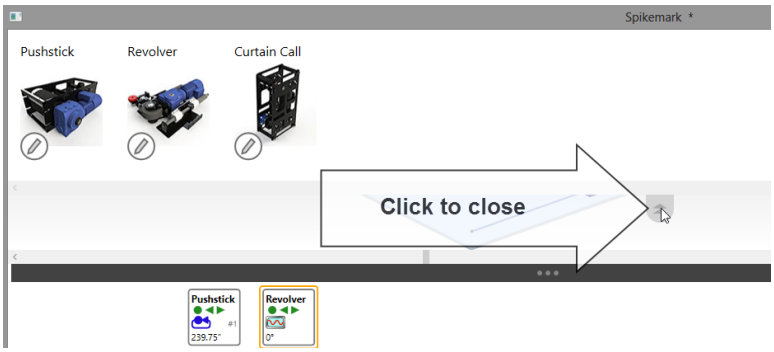


Figure 154

5. Depending on the position of other Pushstick machines in the **Stage Model**, the new Pushstick may be obscured. Adjust the **Schematic** settings to place the new machine on the virtual stage.

6. Change the name and other settings as needed.

Adding a Curtain Call Winch to the Show

1. Click or tap the Machine Library menu (Figure 155).

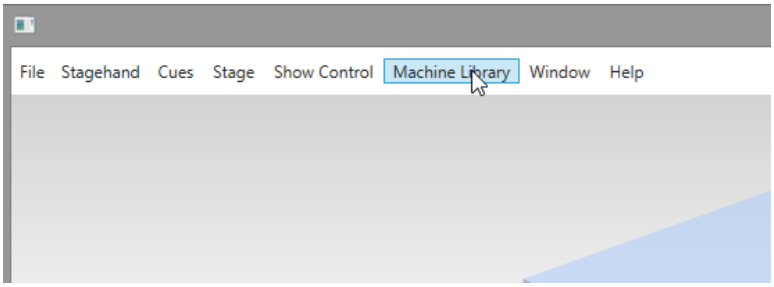


Figure 155

2. The Machine Library panel slides down from the top of the main window (Figure 156).

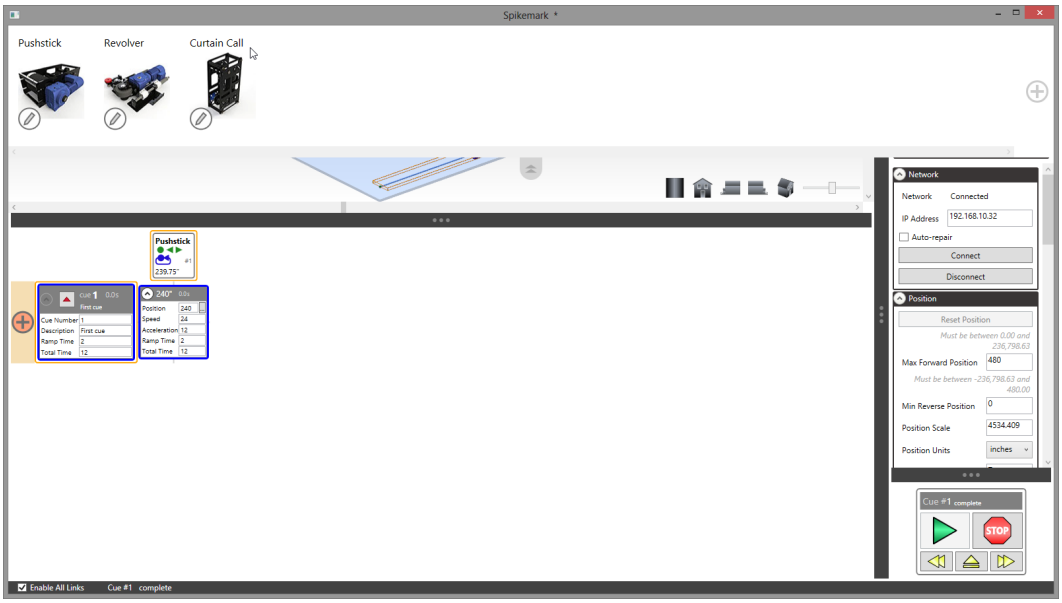


Figure 156

3. Drag the Curtain Call machine from the **Machine Library** and drop it on either on the **Stage Model Viewer** or the **Cue Grid** (Figure 157).

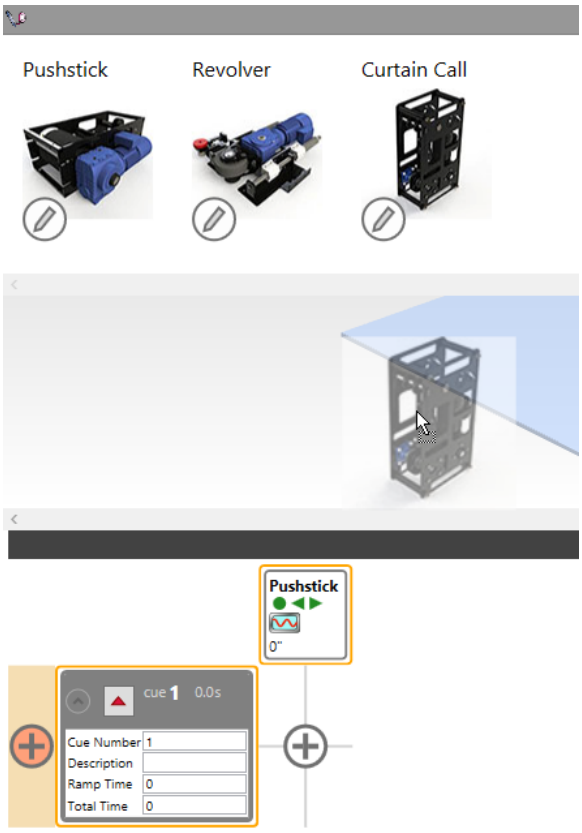


Figure 157

4. To close the Machine Library panel, either press the Esc button or click the Machine Library close tab (Figure 158).

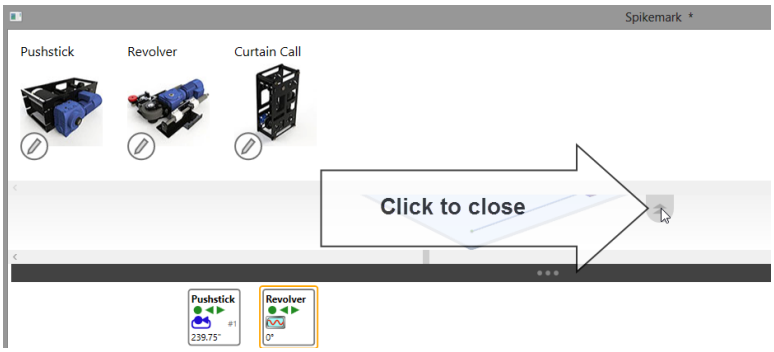


Figure 158

5. Depending on the position of other Curtain Call machines in the Stage Model, the new Curtain Call may be obscured. Adjust the Schematic settings to place the new machine on the virtual stage.

6. Change the name and other settings as needed.

Adding a Revolver to the Show

1. Click or tap the Machine Library menu (Figure 159).

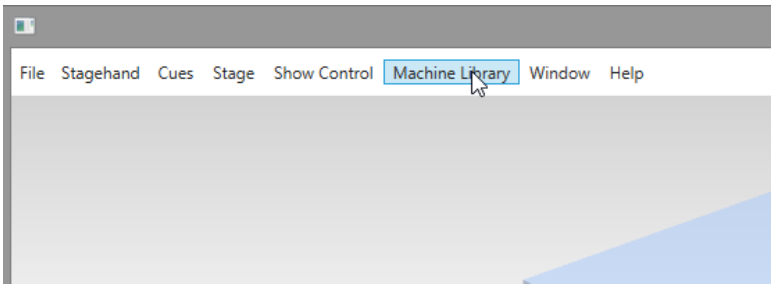


Figure 159

2. The Machine Library panel slides down from the top of the main window (Figure 160).

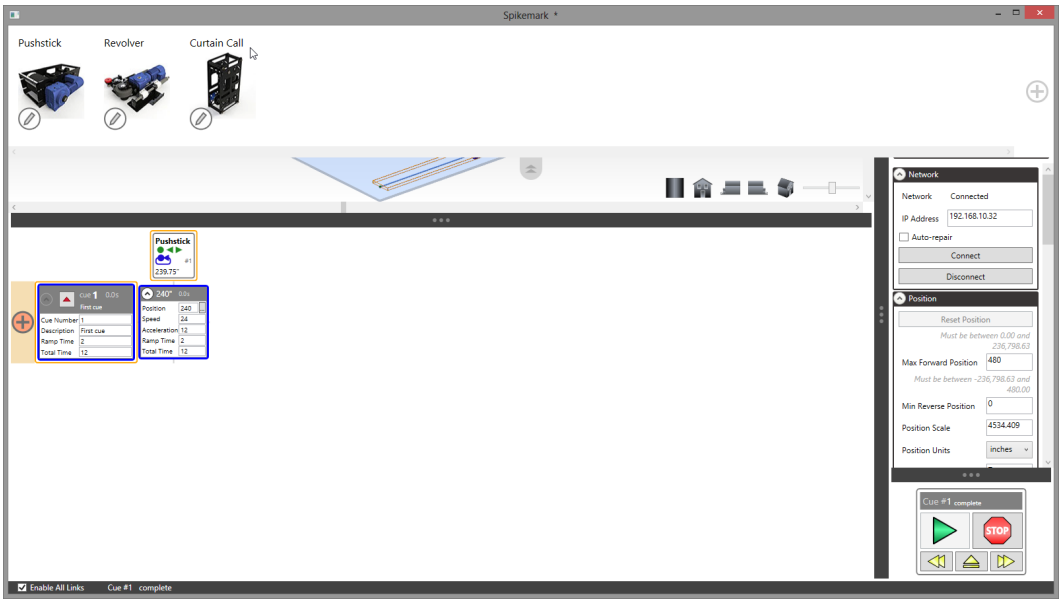


Figure 160

3. Drag the Revolver machine from the Machine Library and drop it either on the Stage Model Viewer or the Cue Grid (Figure 161).

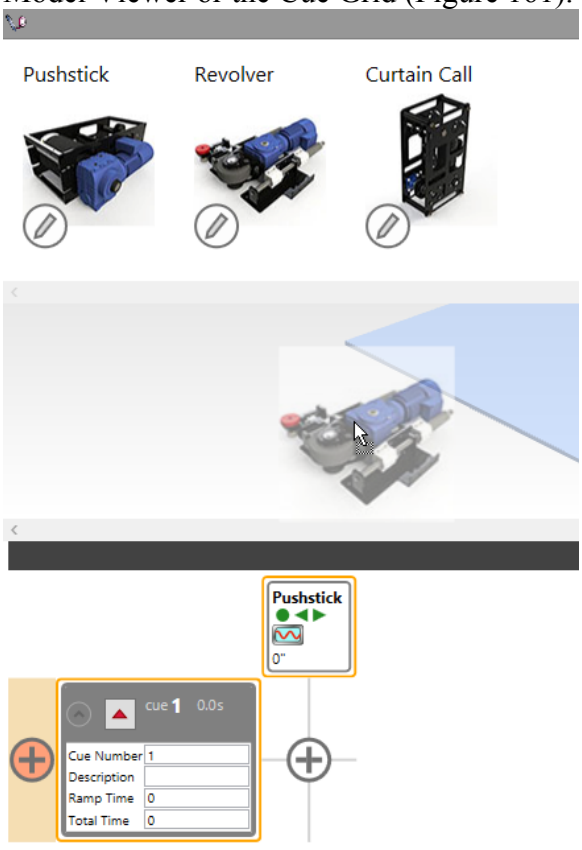


Figure 161

4. Because the Position Scale of a Revolver is dependent on the diameter of your turntable deck, Spikemark prompts you for the diameter of your turntable (Figure 162).

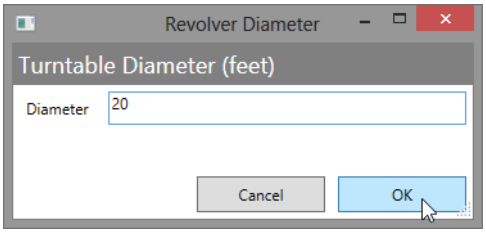


Figure 162

NOTE: The Position Scale that Spikemark computes for a Revolver machine is based on a theoretical ideal diameter of your turntable and the Revolver's encoder wheel. The initial Position Scale will likely need some adjustment to be accurate for your show. See *Setting Position Scale for a Turntable* on page 74 for details.

5. Enter in the diameter of your turntable deck in feet (Figure 162).

6. To close the Machine Library panel, either press the Esc button or click the Machine Library close tab (Figure 163).

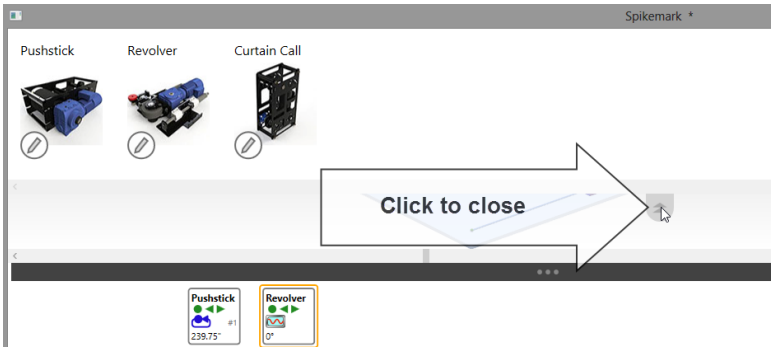


Figure 163

7. Depending on the position of other Revolver machines in the Stage Model, the new Curtain Call may be obscured. Adjust the Schematic settings to place the new machine on the virtual stage.

8. Change the name and other settings as needed.

Adding a Custom Machine to the Machine Library

1. From the Stagehand menu select Add Motor and adjust all of the configuration parameters in the Stagehand Editor Pane that you want to be included in the Machine Library template (Figure 164).

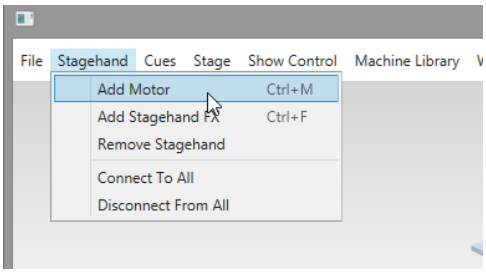


Figure 164

2. Select the machine that you just configured (Figure 165).

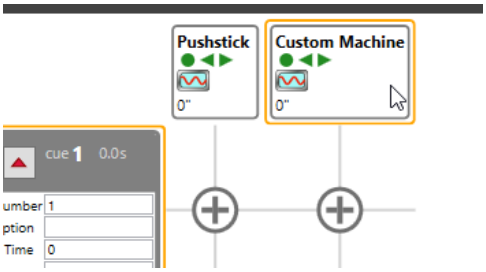


Figure 165

3. Click or tap the Machine Library menu.
4. The Machine Library panel slides down from the top of the main window.
5. Click the “+” button on the far right side of the Machine Library panel (Figure 166).

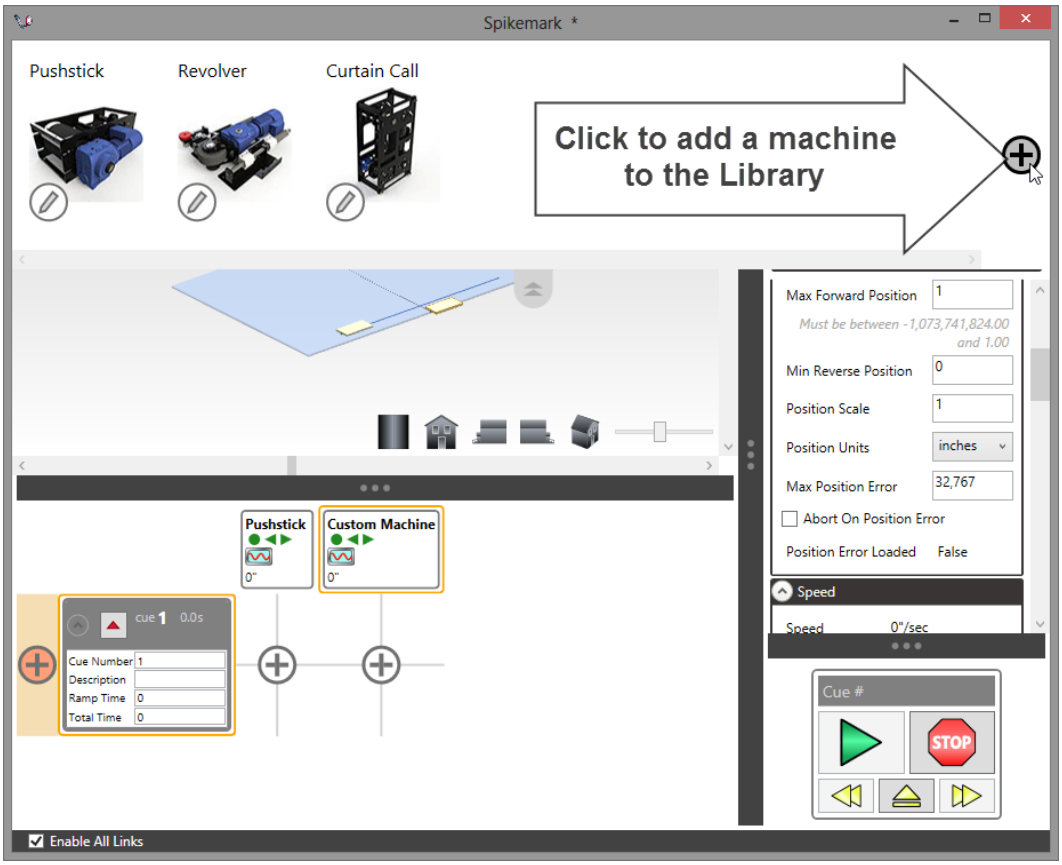


Figure 166

6. Click the Edit button to change the name of the Machine Library item (Figure 167).

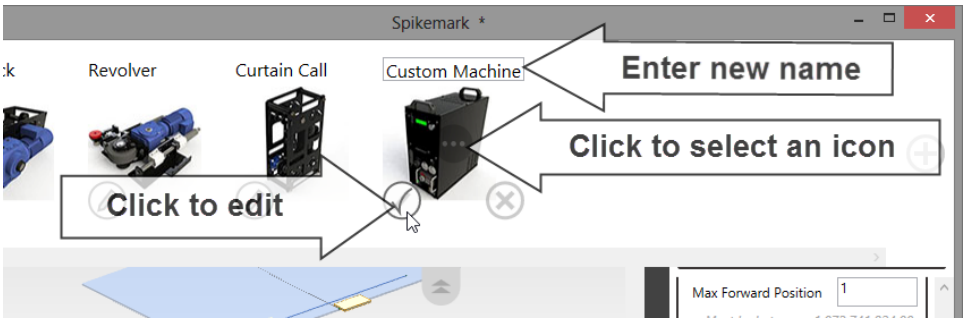


Figure 167

7. Click the center of the image to change the thumbnail for the Machine Library item.

8. Click the Edit button again to finish editing the Machine Library item.

Deleting a Machine from the Machine Library

1. Click or tap the Machine Library menu.
2. The Machine Library panel slides down from the top of the main window.

3. Click the Edit button on the item you want to delete from the Machine Library.
4. A Delete button will appear in the lower right corner of the thumbnail. Click the Delete button (Figure 168).

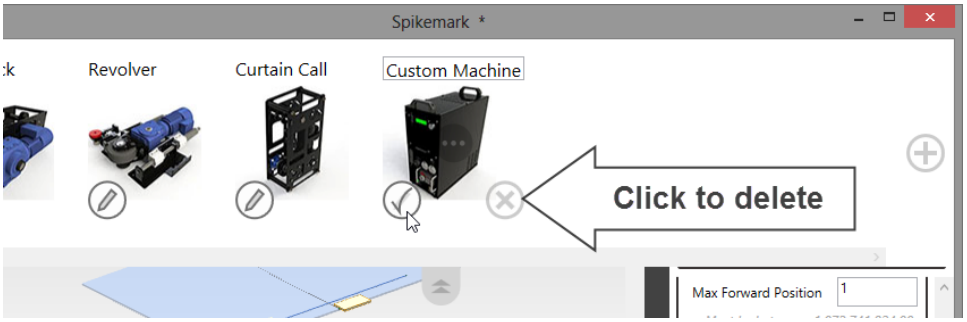


Figure 168

5. Click **Yes** on the **Delete Machine** dialog box (Figure 169).

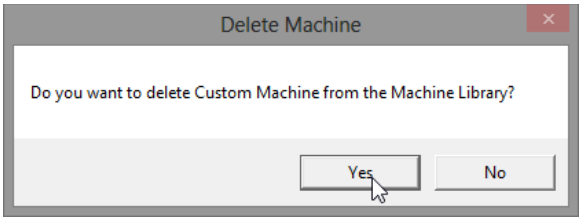


Figure 169

The Stagehand FX

A Stagehand FX (or FX) is an input/output device that has a collection of parameters which are typically set once during a production. It differs from a Stagehand *Motor*, which controls the position of a motor, whereas the Stagehand FX turns any device (motor, valve, etc.) on or off. The **Stagehand Editing Pane** is on the right side of the Spikemark screen and allows you edit all of the FX configuration parameters for the show (Figure 170).

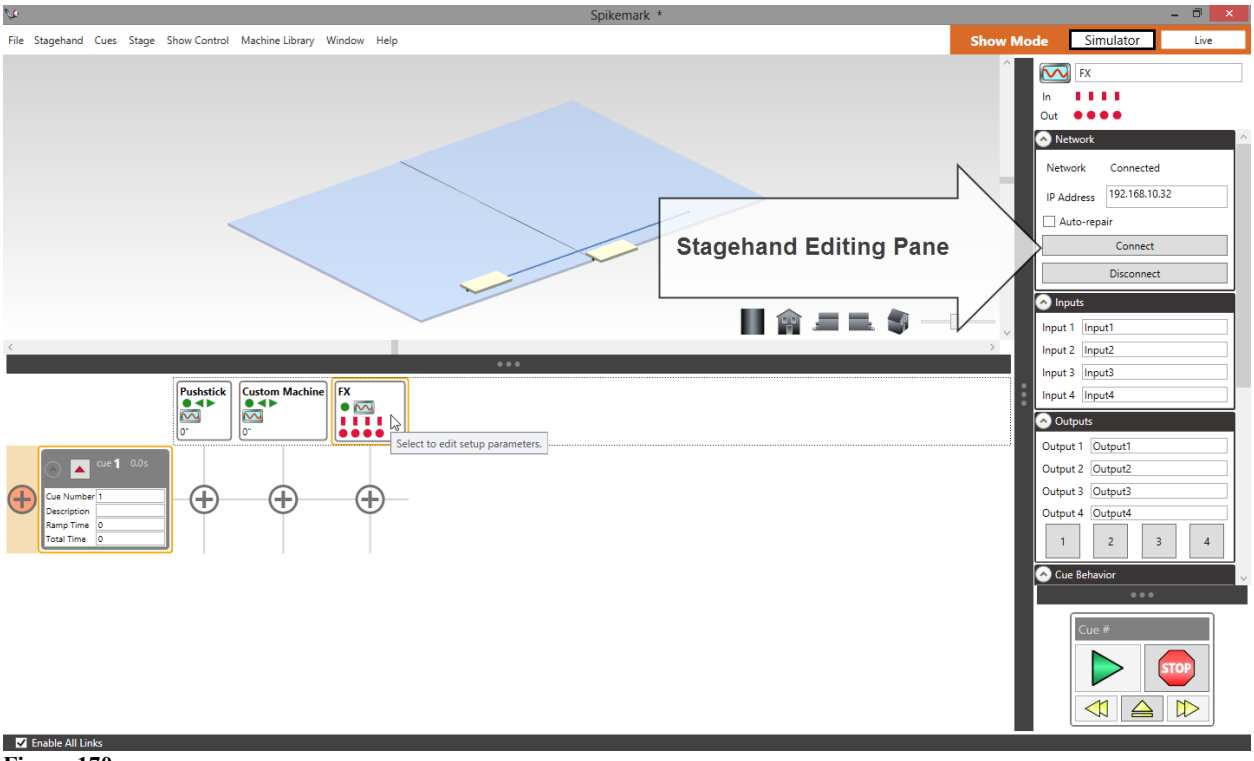


Figure 170

Adding a New Stagehand FX

1. Right-click in the **Cue Grid** and select **Add Stagehand FX** (Figure 171).

Figure 171

Or, in the **Stagehand** menu, click **Add Stagehand FX** (Figure 172).

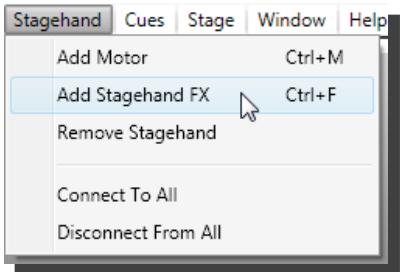


Figure 172

Or, press **Ctrl + F.**

Stagehand FX Configuration

1. Select the Stagehand FX you wish to update in the **Cue Grid**.
   The selected Stagehand FX will be indicated with an orange outline (Figure 173).

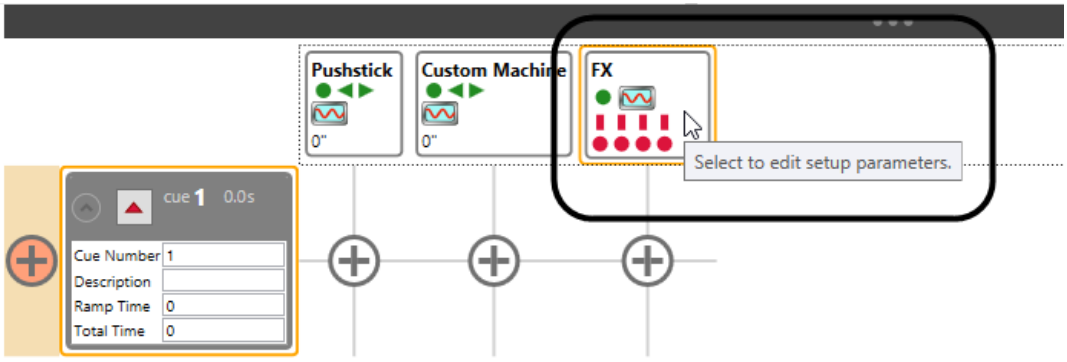


Figure 173

2. Once you have selected an FX in the **Cue Grid**, you can begin editing its parameters in the **FX Editing Pane,** the scrolling window on the right side of the Spikemark screen that contains sections of parameters grouped by function. You can collapse or expand these sections to show or hide the information (Figure 174).

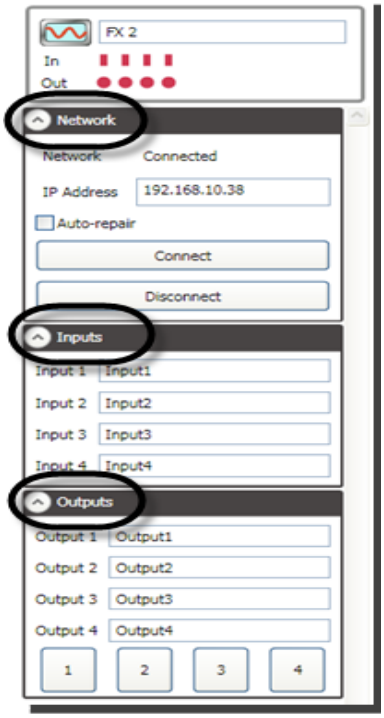


Figure 174

Renaming an FX

To help identify all the Stagehand FX devices in your show, you can give each FX a unique name. Additionally, you can rename all of the Inputs and all of the Outputs for an FX to help remember the purpose of each switch in the Stagehand FX.

1. Select the Stagehand FX to be edited in the **Cue Grid** and then select the field that contains the default name.
   In Figure 175, the default name is "FX2."

Figure 175

2. Type a new name into the field and press **Tab** key or click out of the field to update the name in the **Cue Grid**.
   In Figure 176, the new name is "Trap Door."

Figure 176

Renaming FX Inputs and Outputs

1. Select the Stagehand FX to be edited in the **Cue Grid**

2. In the **Inputs** section of the FX Editing Pane, enter the new **Input** name(s) and in the **Outputs** section of the FX Editing Pane, enter the new **Output** name(s) (Figure 177).

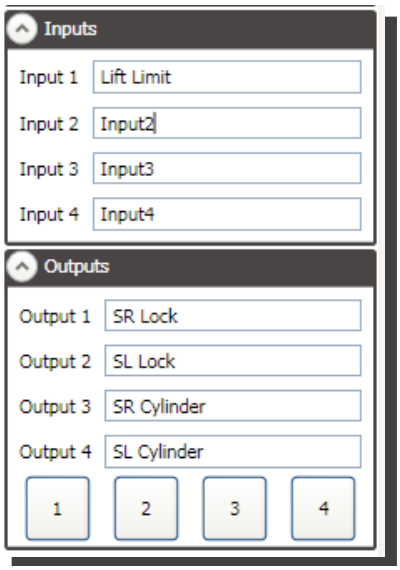

Inputs
Input 1
Lift Limit
Input 2
Input2
Input 3
Input3
Input 4
Input4
Outputs
Output 1
SR Lock
Output 2
SL Lock
Output 3
SR Cylinder
Output 4
SL Cylinder
1
2
3
4

Figure 177

FX Status

Stagehand FX Status information is displayed next to the Stagehand FX name in the **Stagehand FX Editing Pane** and in the **Cue Grid** column headers. The status of the Stagehand FX device is indicated by various icons. Each icon status is described below.

Icon	Status
	FX disconnected from the network.
	FX connected to the network, but idle.
	FX is inactive and won't run cues or affect cue links.
	FX has completed a cue.
	Emergency stop.
	Cue loaded.
	Soft stopping, but not yet fully stopped.
	Stopped mid-cue by a soft-stop command.
	An output is active in manual mode.
	One (or more) output(s) is running in a cue.
	All outputs have been turned off because of a fault condition.

Manually Activating FX Outputs

If you want to test a Stagehand FX output switch, you can use the manual output buttons in the Stagehand FX Editing Pane. Press the button that corresponds to the output you wish to activate. The button behaves like a momentary switch; the FX output will be active as long as the button is held down. When you release the button, the FX output will deactivate.

1. Select the Stagehand FX you wish to activate in the **Cue Grid**.
2. Click and hold down one of the output buttons in the **Outputs** section of the **FX Editing Pane** (Figure 178).

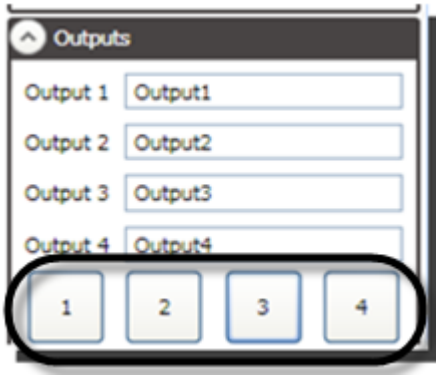


Figure 178

3. Release the button to deactivate the output.

Setting the FX Active Mode

Most of the time, you will want the Stagehand FX to load and run cues normally. However, there are occasions when an effect needs to be temporarily removed from the show. Perhaps there is a problem with the scenery on a winch track, or perhaps you need to run through movements using only the Stagehands downstage and don't want anything upstage to move.

1. Select the Stagehand FX you wish to disable temporarily in the **Cue Grid**.
2. In the **Cue Behavior** section of the **Stagehand FX Editing Pane**, deselect the **Active** field (Figure 179).
   *The FX will not run cues until the **Active** field is selected again.*

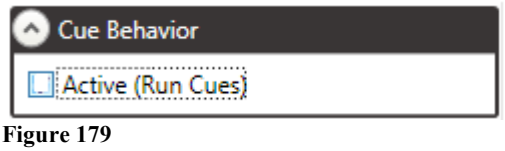


Figure 179

3. You can also set the Active mode by right-clicking on the Cue Grid header as shown in Figure 180.

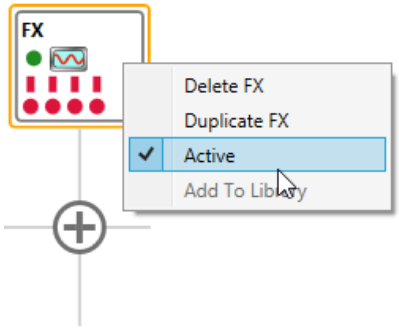


Figure 180

NOTE: When a Stagehand FX is deactivated (not Active), it will not affect the operation of cue *links* (see "Cue Links" on page 134). Normally a cue needs to successfully load all motor and FX movements before it can be executed by a cue link. However, if a Stagehand FX within in a cue is *not* Active—and thereby unable to turn a motor on or off—the entire cue will *still* be allowed to execute. Spikemark will trust that *you have ascertained that it is safe for the cue link to run without this FX* and that you are overriding the normal requirement that all FXs must be able to turn a motor on or off before a linked-cue sequence initiates.

EXAMPLE: Assume you have written Cue 1, in which a LIGHT [controlled by a Stagehand FX] goes on.

Also assume you have written Cue 2, in which a TAB CURTAIN [controlled by a Stagehand FX] opens and WAGON A [controlled by a Stagehand Motor] moves down stage from behind the curtain to in front of the curtain.

Now assume that Cue 1 and Cue 2 are linked—by position—so that the LIGHT goes ON when WAGON A travels to position 60".

Lastly, assume that you want to rehearse WITHOUT the TAB CURTAIN.

To do this, you would open the TAB CURTAIN, to make sure it is out of the way of WAGON A. Next, you would deactivate the TAB CURTAIN [by deselecting the **Active** field in the Cue Behavior section of the tab curtain's FX Editing Pane]. Then, it would then be safe to test MOTOR A and the LIGHT, as you've moved the curtain up and out of the way of MOTOR A.

> Deactivating the FX does not stop the links between the cues. The link between Cue 1 and Cue 2 will still work [in this example, the LIGHT will go ON once MOTOR A has reached the 60" mark].

The **Active** state of a Stagehand FX can be controlled through any FX input action. Using an input on a Stagehand FX, you can quickly deactivate one or many Stagehand Motors or FXs. See the "FX Input Action" section on page 112 for more information.

FX Input Action

Stagehand FX Inputs can be used in a couple of different ways to help create complex cue sequences. In the *"Linking Cues by FX Input"* section on page 139, you'll learn how to use FX Inputs to fire cue links, but here we will illustrate another way to use FX Inputs: Input Actions.

Input Actions introduce a way to trigger events in the show that are not directly related to a cue. FX Input Actions give you a way to tell Spikemark to do something whenever an input is activated, regardless of what cue is running. We are starting modestly with two action types, but look for more actions in future releases.

Deactivating Stagehand FX Action

You can define an Input Action that will toggle the **Active** state of one or more Stagehand Motors *or* FXs to temporarily keep them from running cues during the show. This may be handy if you have a scenic element that is involved in a number of cues, but requires tight synchronization with an actor's movements. Some nights the computer sequencing works fine, but some nights you need to take manual control of the motor and jog it to match the actor. When that happens, you don't want to mess up all the other cues in the sequence, so you wire an "Override" button into a Stagehand FX and configure it as illustrated in the steps below.

1. Select the Stagehand FX you wish to configure in the **Cue Grid**.

2. In the **Input (1, 2, 3 or 4) Action** section of the **FX Editing Pane**, click the **Add Action** button (Figure 181).

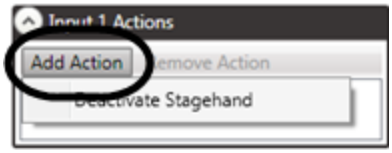


Figure 181

3. Select **Deactivate Stagehand** as the action type (Figure 182).

Figure 182

4. From the **Deactivate Stagehand** drop down menu, select the FX or Motor that should be deactivated when the FX Input is closed (Figure 183).

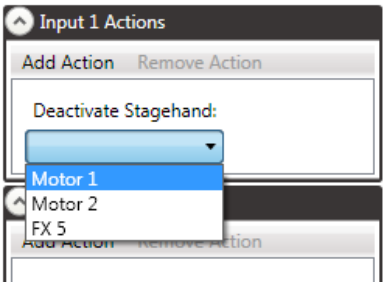


Figure 183

5. Repeat as necessary if you would like a single FX Input to deactivate multiple Stagehands—Motors *or* FXs (Figure 184).

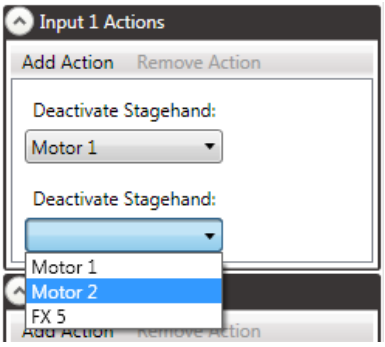


Figure 184

Soft Stopping FX Action

There are times when you may need to soft stop a device in a show based on actor timing. For example, maybe you have two concentric turntables, which both start moving at the same time, but one which needs to stop when an actor says a particular line. In this example, you would not want to set up a cue in which the turntable reached a target position, but, instead, you could use an FX input and set it so that when manually activated after the actor spoke the line, the motor would slow down to its programmed deceleration rate and then stop.

1. Select the Stagehand FX you wish to configure in the **Cue Grid**.

2. In the **Input (1, 2, 3 or 4) Action** section of the **FX Editing Pane**, click the **Add Action** button (Figure 185).

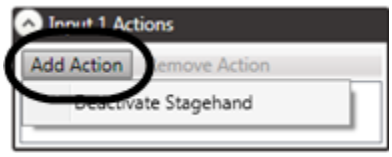


Figure 185

3. Select **Soft Stop Stagehand** as the action type (Figure 186)

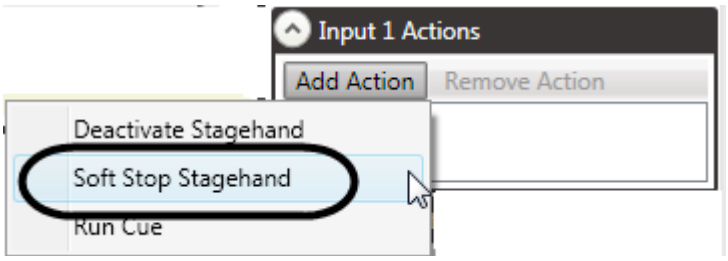


Figure 186

4. From the **Soft Stop Stagehand** drop down menu, select the FX or Motor that should be soft stopped when the FX Input is closed (Figure 187).

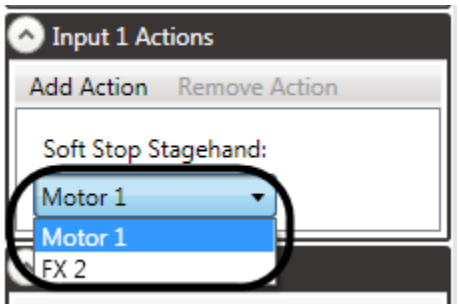


Figure 187

5. Repeat as necessary if you would like a single FX Input to deactivate multiple Stagehands—Motors *or* FXs (Figure 188).

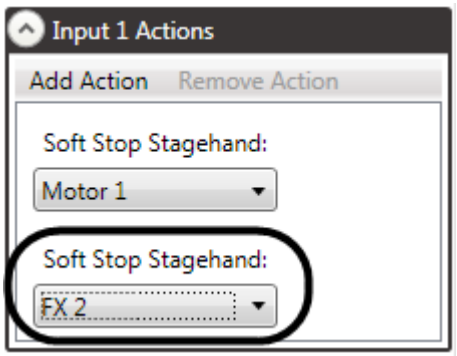


Figure 188

Run Next Cue FX Action

In some environments, particularly interactive exhibits, you may want Spikemark to run through a series of cues in sequential order by repeatedly pressing a single button. For instance, a presenter giving an interactive lecture and may need to execute a motion sequence of motorized props in an exhibit. In such a case, Spikemark can be used with a Stagehand FX, a button wired into an input, and a Run Next Cue action. When the presenter needs to move to the next sequence, she simply presses the button connected to Stagehand FX and the exhibit advances. While this action is rarely useful in live theatre, it does find use in non-traditional applications.

To add a Run Next Cue FX action:

1. Select the Stagehand FX you wish to configure in the Cue Grid.
2. In the Input (1, 2, 3, or 4) Action section of the Stagehand Editing Pane, click the Add Action button (Figure 189).
3. Select Run Next Cue as the action type (Figure 189).

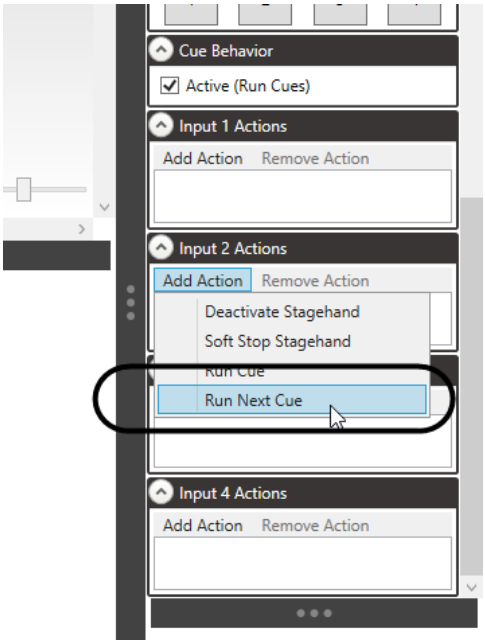


Figure 189

NOTE: The Run Next Cue action will effectively create an always-armed GO button. This is not intended for use where the motorized effects could pose a threat to human safety.

Removing a Stagehand FX Input Action

1. Select the Stagehand FX for which you wish to remove the Input Action in the **Cue Grid**.
2. In the **Input (1, 2, 3 or 4) Action** section of the **FX Editing Pane**, highlight the desired action and press **Remove Action** (Figure 190)**.**

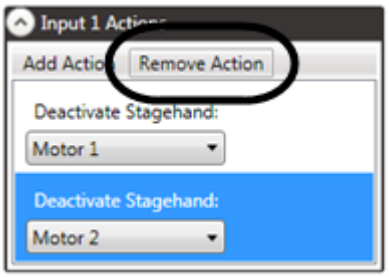


Figure 190

Network Connections

Editing Stagehand Network Settings

The **Network** section of the **Stagehand Motor** or **FX Editing Pane** (Figure 191) contains the settings that pertain to the Ethernet connection between Spikemark and the Stagehand Motor or FX. The fields in the **Network** section are described below:

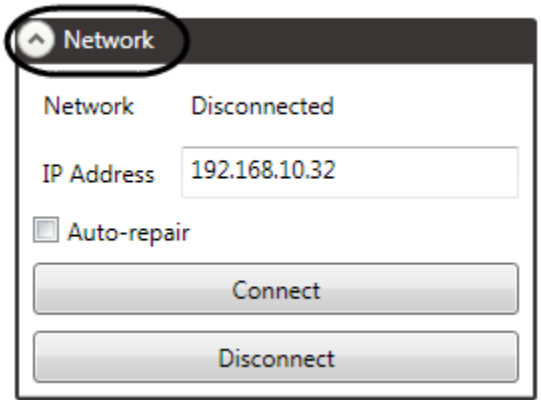


Figure 191

The **Network** field (Figure 192) shows the current status of the network connection. The possible status values are:

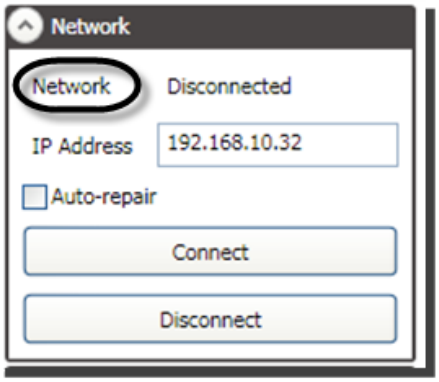


Figure 192

Disconnected: The network connection has not been made yet.

Connecting: A connection is trying to be established.

Errors: The connection failed.
Connections can fail for various reasons, but commonly: a cable was removed, the specified Stagehand address is wrong, or the computer running Spikemark has an incompatible

address (see the "Setting Up a Network Connection" section on page 41 for steps to configure your computer's IP address).

The **IP Address** field (Figure 193) contains the IP Address of the selected Stagehand Motor or FX that you wish to connect to. Enter the **IP Address** exactly how it is displayed on the Stagehand Motor or FX face panel. Valid address range from 0.0.0.0 to 255.255.255.255, but remember that the first three segments of the address must match between the computer running Spikemark and all the Stagehands on the network, while the last segment must be unique for each device.

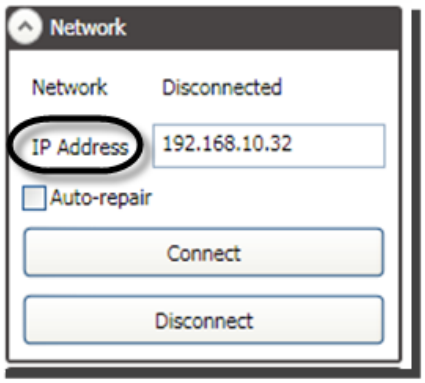


Figure 193

If the **Auto-repair** (Figure 194) field is selected, Spikemark will automatically try to reconnect to a Stagehand Motor or FX if the connection is lost for any reason. Most of the time you'll turn this option on for running shows, but you may want to turn it off during initial set up to take a motor 'offline'.

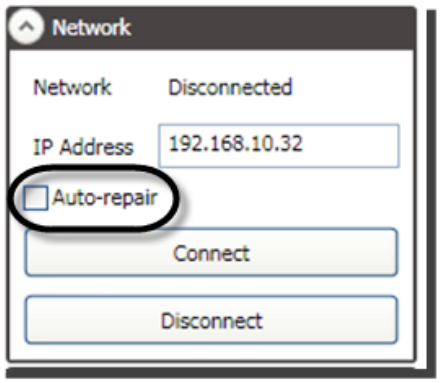


Figure 194

Clicking the **Connect** button (Figure 195) will begin connecting to the Stagehand Motor or FX.

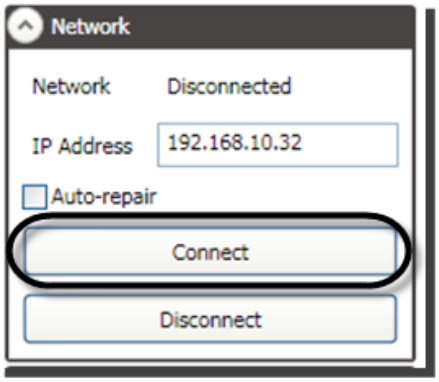


Figure 195

Click the **Disconnect** button (Figure 196) to disconnect from the Stagehand Motor or FX.

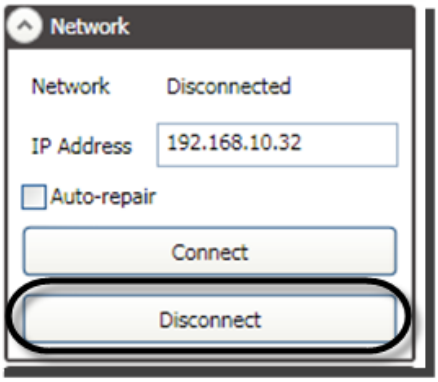


Figure 196

Connecting to a Stagehand Motor or FX

All communications between Spikemark and different Stagehand Motors or FXs happen over the network. To run any cues, move any motors, or activate an FX output, a network connection must be made to the Stagehand Motor or FX you wish to control. The connection only needs to be established once per show, unless the network connection is lost unexpectedly. There are three ways to establish a network connection.

1. Select the motor or FX you wish to connect to from the **Cue Grid**.

2. In the **Network** section of the **Stagehand Editing Pane**, click the **Connect** button (Figure 197).

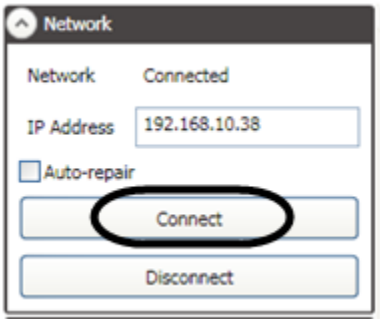


Figure 197

Or, you can connect to all the Stagehand Motors and FXs in the show at once by clicking **Connect To All** in the **Stagehand** menu (Figure 198).

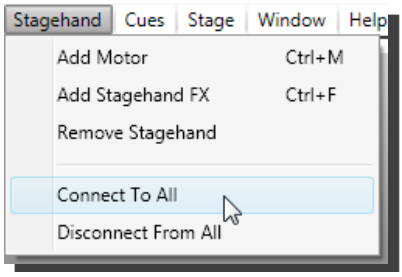


Figure 198

Or, each time you open a show file, Spikemark will automatically attempt to connect to all the Stagehand Motors and FXs in the show.

Disconnecting from a Stagehand Motor or FX

You can break the network connection to a Stagehand Motor or FX any time you need to temporarily remove it from Spikemark's control. Disconnecting from the Stagehand Motor or FX will turn off all position & status updates from the Motor or FX. If you would like to *temporarily* remove the Motor or FX from the show, but maintain network communication, see the "Setting a Motor's Active Mode" section on page 88 or the "Setting the FX Active Mode" on page 109.

NOTE: Make sure you deselect the Auto-repair **field in the Network section of the Stagehand** Motor **or** FX Editing Pane **if you want to disconnect from a motor or FX or Spikemark will immediately try to reconnect.**

There are two ways to break a network connection.

1. Select the motor or FX you wish to disconnect in the **Cue Grid**.

2. Press the **Disconnect** button in the **Network** section of the **Stagehand Motor** or **FX Editing Pane** (Figure 199).

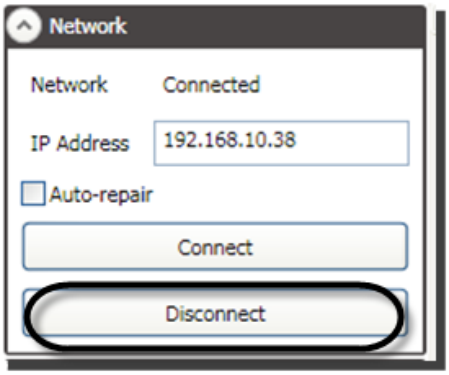


Figure 199

Or, to disconnect from *all* Stagehand Motors or FXs in the show, in the **Stagehand** menu, click **Disconnect From All** (Figure 200).

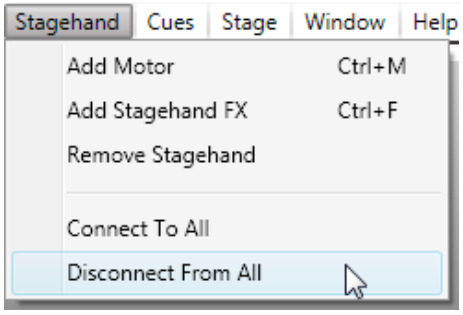


Figure 200

Writing Cues

A Cue is a collection of movements including any number of motors or FX outputs that all start when you press the **GO** button on the **Showstopper** or on the **Cue Controller**. You can have an unlimited number of cues to your show. Each cue has a cue number that can be any decimal number, allowing for 'point cues' (for example: 1.5, 209.52, etc.). Spikemark displays all the cues in the show in the **Cue Grid** as rows sorted by Cue Number. When **Spikemark** is first launched, a single cue, Cue #1, is created by default (Figure 201). You can use this cue as a starting point or delete it and start with a blank slate.

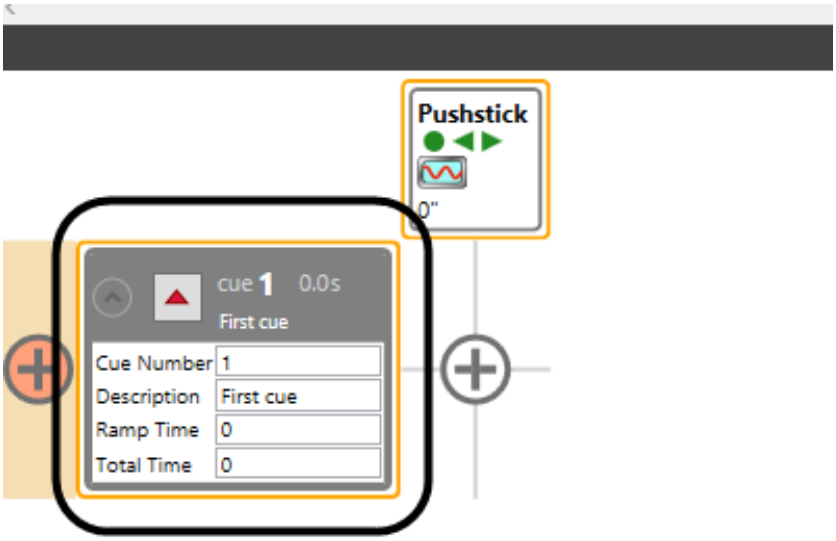


Figure 201

NOTE: You can write cues with or without being connected to any Stagehand Motors or FXs. You can "rough in" all of your cues before setting up your motors on stage. By using the Spikes feature and a scaled ground plan, you can write all of the cues for the show and then tweak final positions when the scenery is installed (see the "Spikes" section on 163).

Adding Cues

There are three ways to add cues to your show.

1. Right-click on a blank area in the **Cue Grid** and select **Add Cue** (Figure 202)**.**

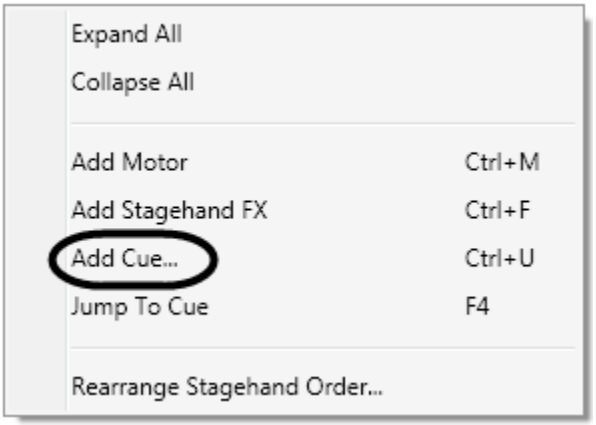


Figure 202

Or, in the **Cues** menu, click **Add Cue** (Figure 203).

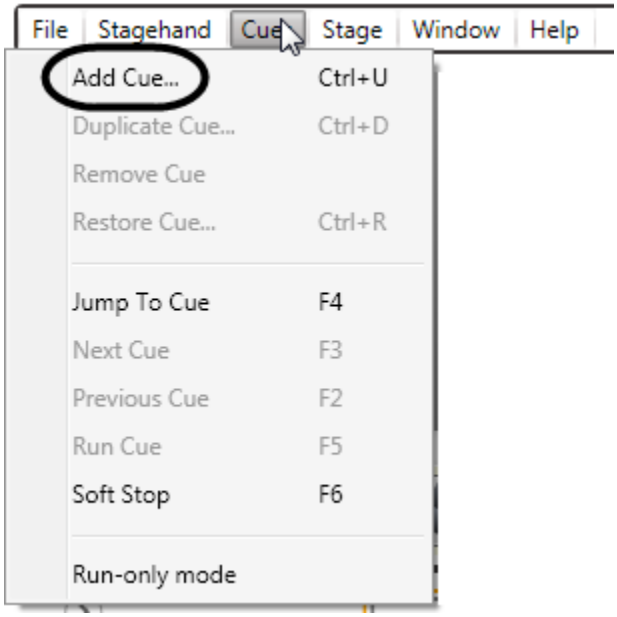


Figure 203

Or, use the keyboard shortcut **Ctrl** + **U**.

2. A dialog appears prompting you for a **Cue Number** and **Cue Description** (Figure 204).

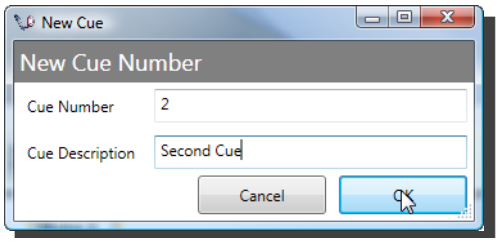


Figure 204

NOTE: You can edit the Cue Number **or accept the default. The** Cue Description **is solely for your benefit. You can enter a descriptive name to help identify the cue in the** Cue Grid**.**

3. The newly created cue will appear in the **Cue Grid** (Figure 205).

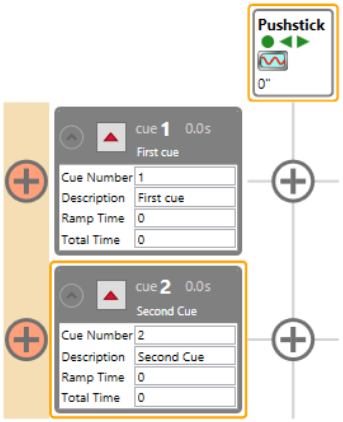


Figure 205

Adding Motor Movements to a Cue

1. Press the "+" button that is displayed at the intersection of the cue row and the motor column (Figure 206).

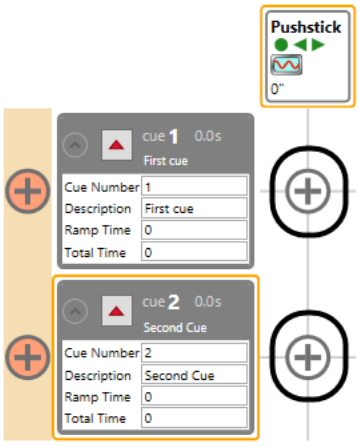


Figure 206

2. After you press the "+" button, a movement will appear displaying the parameters for the motor in the cue (Figure 207).

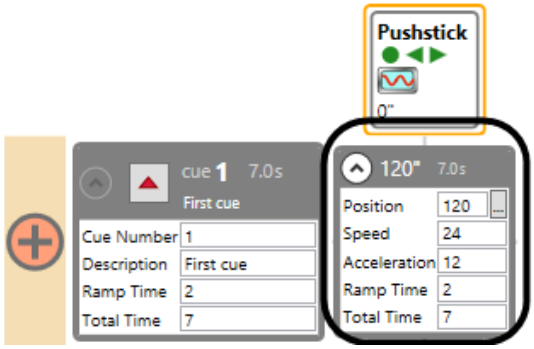


Figure 207

Motor Movement Parameters

Position (Figure 208) is the target position where you want the motor to go. This number is displayed in scaled units (inches, feet, degrees, etc.). By default, Spikemark uses the current position of the motor as the target position. The button to the right of the **Position** text box will bring up the **Spike Selection** window. Spikes are discussed in the "Spikes" section on page 163.

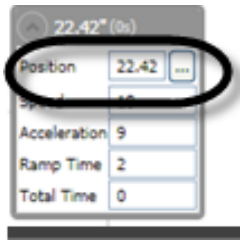


Figure 208

Speed (Figure 209) indicates the speed at which you want the motor to travel. This number is displayed in units per second (inches/sec, degrees/sec, etc.). You can synchronize multiple motors by programming them at the same speed, regardless of gearing and encoder differences. Valid speed values are greater than zero and no greater than the **Max Speed**, as defined in the **Speed** section of the **Stagehand Editing Pane**. Spikemark initially uses the **Default Speed**, also defined in **Speed** section of the **Stagehand Editing Pane**.

Figure 209

Acceleration (Figure 210) indicates the rate at which the speed will increase as the motor moves. This number is displayed in units per second per second (inches/sec/sec, etc.). A higher value will cause the motor to accelerate more quickly giving the motion more snap at the beginning and end of the cue. A lower value will cause the motor to accelerate more slowly and smoothly. Valid acceleration values range from greater than zero to no greater than the programmed **Speed** value. Spikemark initially uses the **Default Acceleration** defined in the **Speed** section of the **Stagehand Editing Pane**.

Figure 210

Ramp Time (Figure 211) indicates the number of seconds that the motor will take to accelerate up to programmed speed. If you would like to program the cue in time units, you can edit this field rather than using **Acceleration**. Spikemark will automatically calculate a new **Acceleration** based on the **Ramp Time** you enter.

Figure 211

Total Time (Figure 212) indicates the total number of seconds that the movement will take to complete. If you know that the motor needs to move the scenery in 10 seconds, you can

enter a value of "10" here and Spikemark will recalculate the **Speed** and **Acceleration** values to make that happen.

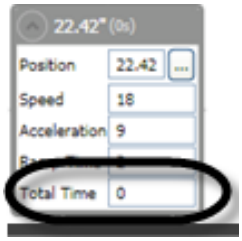


Figure 212

NOTE: Spikemark computes the Total Time and Ramp Time in the first cue of the show by comparing the programmed position to the motor's current position. In all subsequent cues in the show, Spikemark compares the programmed position to the *previous cue's* programmed position.

Changing Motor Position In a Cue

To change the programmed position of a motor in a cue you can do either of the following:

1. Type a new position value directly into the **Position** field (Figure 213).
   *After you enter a new **Position**, Spikemark will recalculate the **Total Time** of the movement based on the new target position. If you had previously specified a **Spike** for the movement it will be removed and replaced with the explicit value entered.*

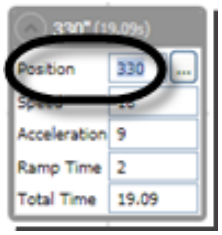


Figure 213

Or, press the **Spike** button next to the **Position** box (Figure 214). Then select a **Spike** from the list and then click **OK** to accept (Figure 215).

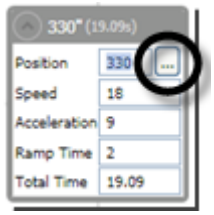


Figure 214

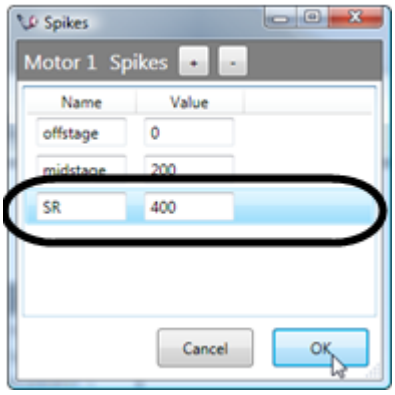


Figure 215

Changing How Fast a Motor Moves Within a Cue

There are two ways to adjust how fast a motor moves: Specify the *speed* of a motor directly or specify *how long* (the time) it should take for the motor to reach the target position and then let Spikemark automatically compute the speed for you.

Specifying the Speed of a Motor Directly

1. Select the motor you wish to update in the **Cue Grid** and select the **Speed** value for the applicable motor movement (Figure 216).

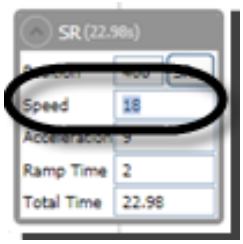


Figure 216

2. Enter in a new **Speed** value and press the **Tab** key. (Figure 217).
   *Note that the **Ramp Time** and **Total Time** are recomputed automatically by Spikemark.*

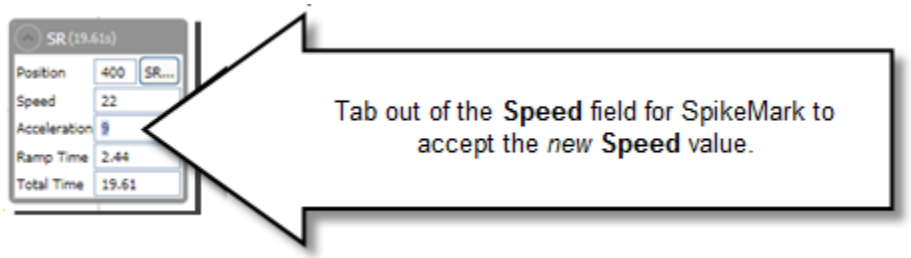


Figure 217

3. Enter in a new **Acceleration** value and press the **Tab** key. (Figure 218).
*Note that the **Ramp Time** and **Total Time** are recomputed automatically by Spikemark.*

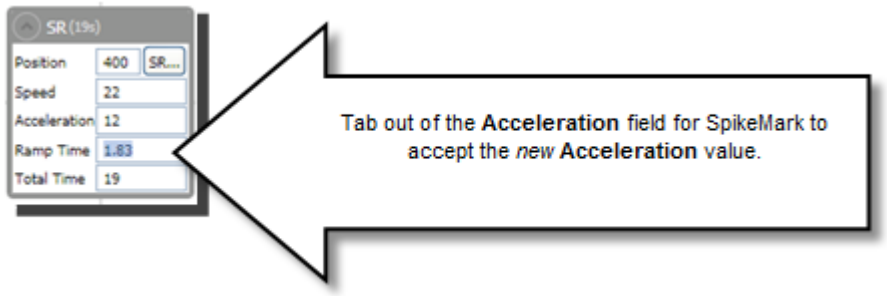

SR (19s)
Position 400 SR...
Speed 22
Acceleration 12
Ramp Time 1.83
Total Time 19
Tab out of the **Acceleration** field for SpikeMark to accept the *new* **Acceleration** value.

Figure 218

Specifying *How Long* the Motor Should Take to Reach the Target Position

1. Select the motor you wish to update in the **Cue Grid** and select the **Total Time** value for the applicable motor movement (Figure 219).

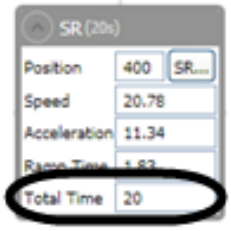


Figure 219

2. Enter in a new **Total Time** value and press the **Tab** key (Figure 220).
*Note that the **Speed** and **Acceleration** are recomputed automatically by Spikemark.*

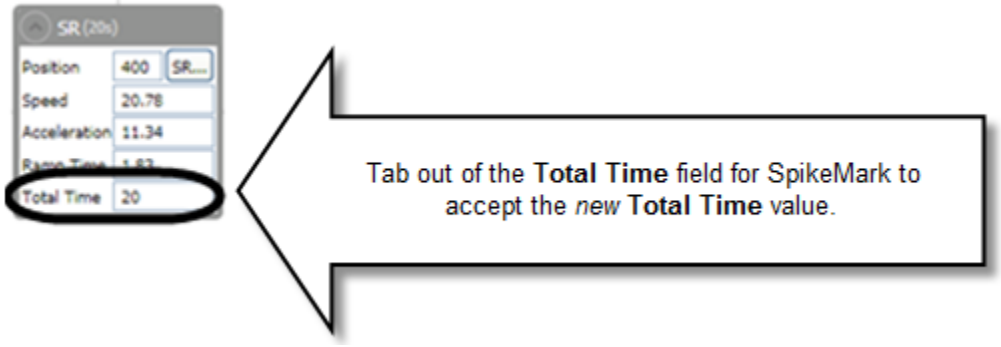

SR (20s)
Position 400 SR...
Speed 20.78
Acceleration 11.34
Total Time 20
Tab out of the **Total Time** field for SpikeMark to accept the *new* **Total Time** value.

Figure 220

3. Select and change the **Ramp Time** value and press the **Tab** key (Figure 221).

*Note that the **Speed** and **Acceleration** are recomputed automatically by Spikemark.*

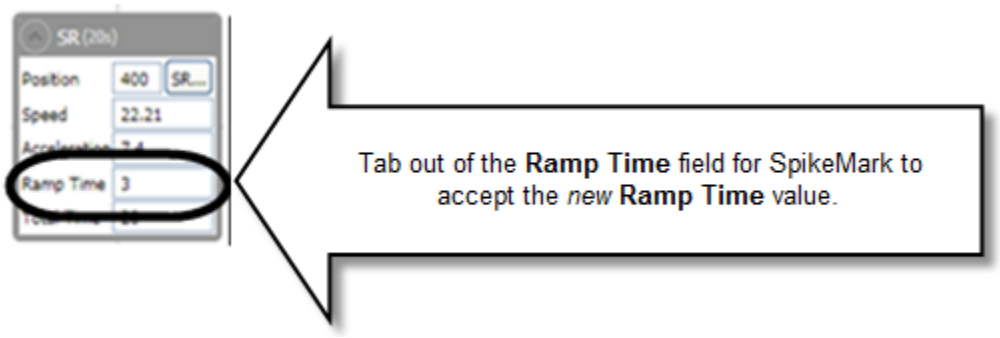


Figure 221

NOTE: If you enter a Total Time or Ramp Time combination that would require the motor to exceed its Max Speed [specified in the Speed section of the Stagehand Editing Pane], then the Total Time and Ramp Time will be recomputed to be *no higher than* the Max Speed.

Changing Cue Time

You can change the time for *all the movements in a cue* by specifying the **Total Time** and **Ramp Time** for the entire cue. This makes it easy to write cues that have all the motors ending at the same time.

1. Select the **Cue** you wish to edit in the **Cue Grid** (Figure 222).

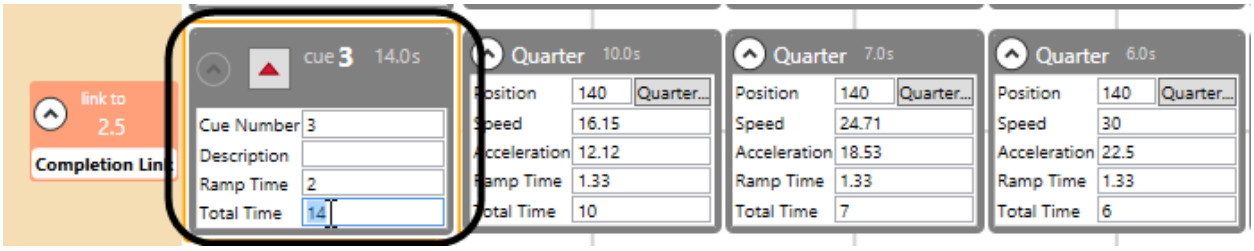


Figure 222

2. Type in the new **Total Time** value you wish to use and press the **TAB** key (Figure 223).
   *Note that all the **Total Time** values for each movement have been updated.*

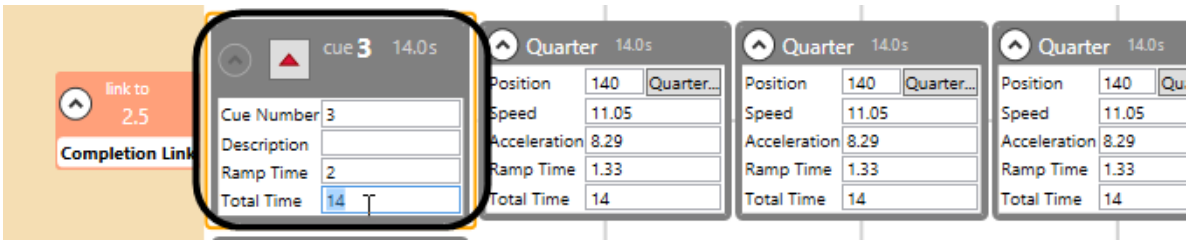


Figure 223

3. Repeat the steps for **Ramp Time**, if desired.

Adding an FX Movement to a Cue

Adding an FX movement to a cue is similar to adding a motor movement.

1. Click the "+" button in the **Cue Grid** correct associated with the Stagehand FX and Cue you wish to edit (Figure 224).

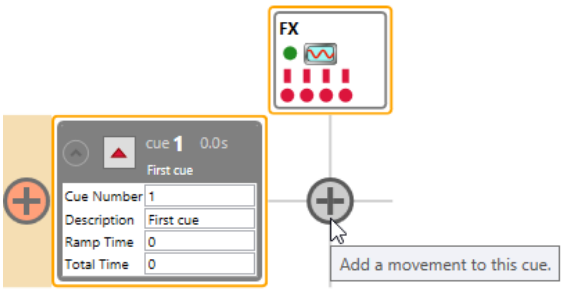


Figure 224

2. An FX movement will appear in the **Cue Grid** (Figure 225).

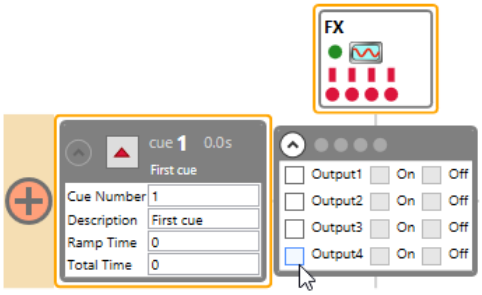


Figure 225

NOTE: An FX movement can turn any output switch on or off in the cue. If a switch is turned on in one cue, it will remain on until another cue turns it off. Therefore, if you don't want to change the state of a switch in a cue, leave the switch unchecked completely.

Figure 226 is an example series of cues that turns Output 1 on, then off, then turns Outputs 2, 3 and 4 on, while leaving Output 1 off.

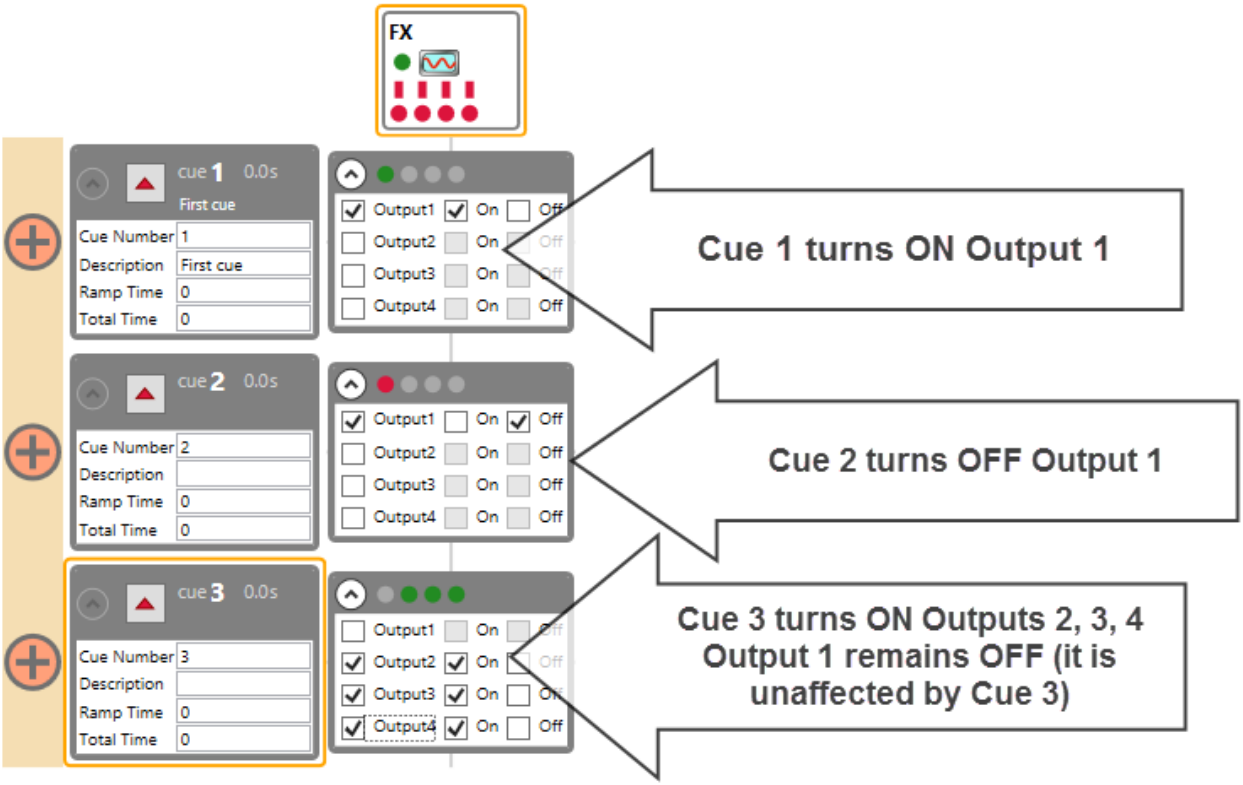


Figure 226

Removing a Motor's Movement from a Cue

Sometimes you will write a cue with a number of motors moving and then discover one of the motor movements should not be in the cue.

1. Right-click on the motor's movement you wish to delete in the **Cue Grid** and select **Delete Movement** (Figure 227).

Figure 227

2. Click **Yes** in the **Remove Movement** confirmation dialog box (Figure 228).

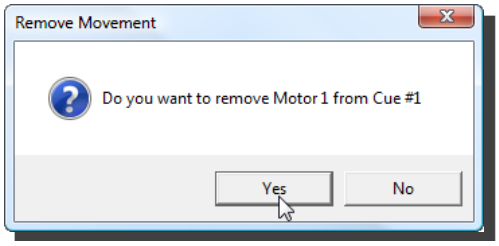


Figure 228

Deleting a Cue

Theatre is a dynamic process. Throughout rehearsals scene shifts will be added, and scrapped as the performance is shaped. To keep up with the rapid changes, Spikemark makes it just as easy to remove cues as it is to add them. There are two simple ways to delete a cue from a show:

1. Right-click on the cue you wish to delete in the **Cue Grid** and select **Remove Cue** (Figure 229).

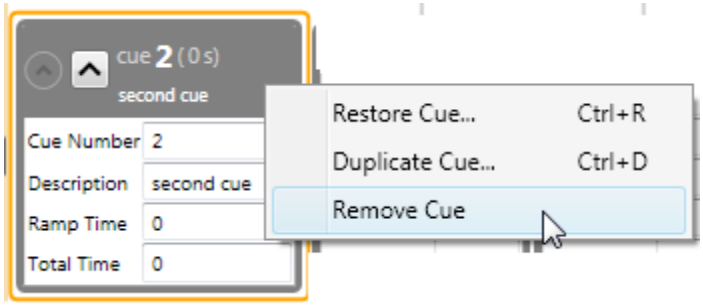


Figure 229

2. Click **Yes** in the **Remove Cue** confirmation dialog box (Figure 230).

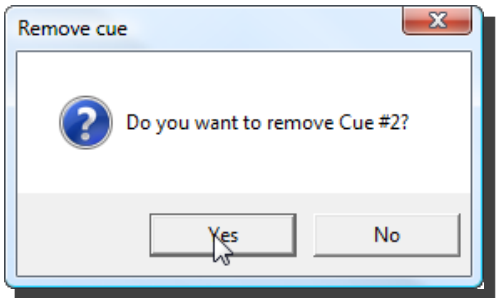


Figure 230

Or, select the cue you wish to delete in the **Cue Grid**. In the **Cues** menu, click **Remove Cue** (Figure 231) and then click **Yes** to confirm the deletion.

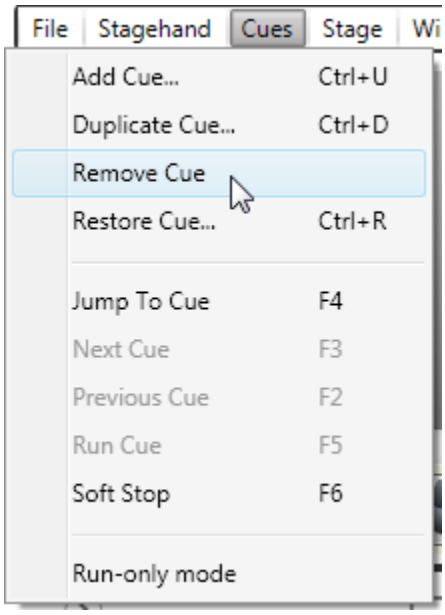


Figure 231

Cue Links

As you are writing cues in a new show, you'll find scene changes in which simple cues fall short of your desired effect. You might want to fire a sequence of movements, all within in one cue, but each with a *different* start time. For example, you might want a two-second delay between the start of each movement. Or, maybe the house wagon needs to wait to move upstage until the balcony unit is clear and in the wings. Whatever the need, links allow you to create sequences of cues that start automatically in a staggered fashion.

Links are great for cue sequences that can be reliably run in the exact same way every performance. If you find that a link is too rigid (for example, you need to run two cues, timed such that the second cue runs right after an actor says a specific line), a cue link might be too rigid, because the timing of the actor's line is unpredictable from night to night. In these instances, you should keep the cues independent or, use **Pile On** cues. Pile On cues are loaded *manually*, and at your own discretion, before the previous cue completes.

When discussing **Cue Links**, we need a vocabulary to refer to the two cues that are linked together:

Parent Cues are the cues being watched by the link. The **Parent Cue** is run by either pressing the GO button, or it may be a Child Cue (see below) of another link in a longer sequence of linked cues. Whenever the Parent Cue is running, the link will check its criteria and automatically start the **Child Cue** when the criteria are met.

Child Cues are the cues that are started by the link. When a link's criteria are satisfied, the link will automatically run the **Child Cue**. Graphically, Child Cues have the link displayed next to them in the **Cue Grid**.

All links are displayed in the left margin of the **Cue Grid** (Figure 232).

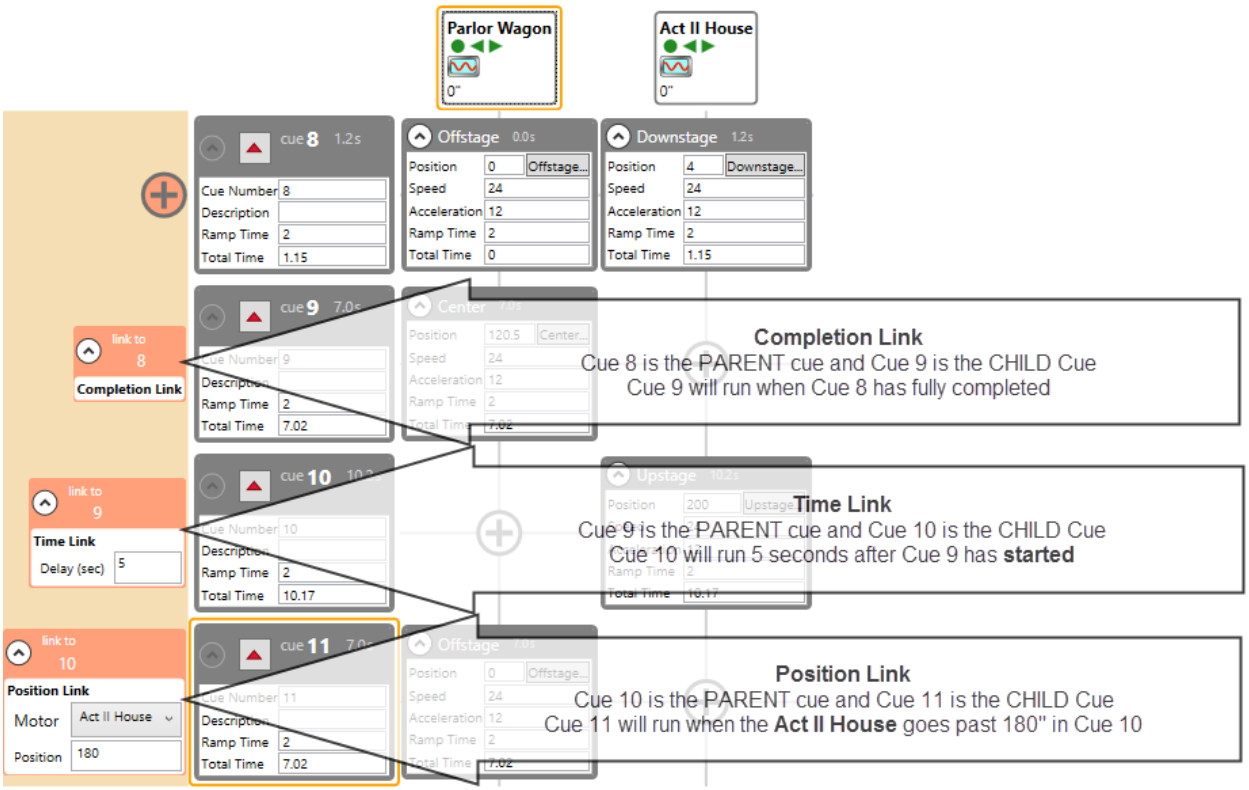


Figure 232

NOTE: You can create an infinite loop by creating two links between two different cues. You would set up your links so that Cue 1 is the Parent Cue of Cue 2 and Cue 2 is the Parent Cue of Cue 1. This might be useful if you are testing machines OR if you need a motor to cycle back and forth indefinitely. To stop such a loop of cues, you would click the Stop button in the Cue Controller, or use the Emergency Stop on the Showstopper or use a "Soft Stop Motor" Input Action with a Stagehand FX.

Linking Cues by Completion

You can link two cues together by completion, so that when the **Parent Cue** completes successfully, the **Child Cue** will begin.

1. Click the **Add Link** button next to the **Child Cue** (Figure 233).

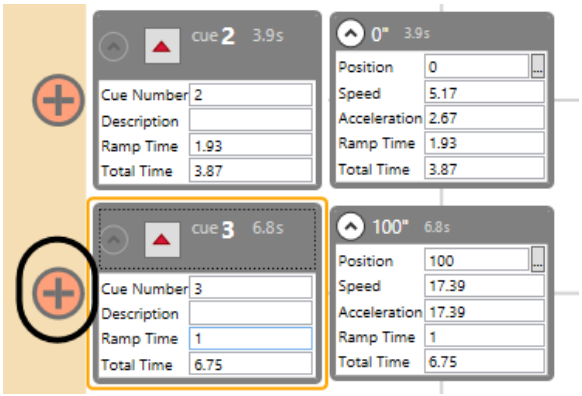


Figure 233

2. Select the **Parent Cue** from the drop down list and select **Completion Link**, then press **OK** (Figure 234).
   *In the example pictured below, the selected **Parent Cue** is "Cue #2."*

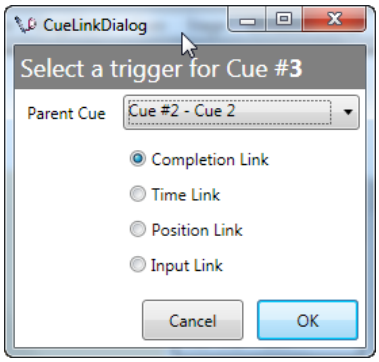


Figure 234

3. The new link will appear in the left margin of the **Cue Grid** (Figure 235).
   In the example below, Cue 3 (Child Cue) will not run until Cue 2 (Parent Cue) has completed.

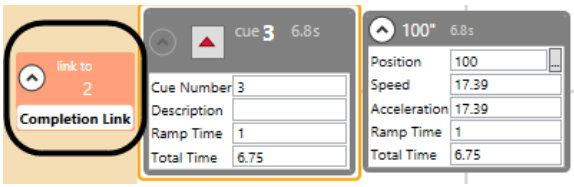


Figure 235

Linking Cues by Time

You can link two cues together with a time delay so that the **Child Cue** runs a programmed number seconds after the **Parent Cue** starts.

1. Click the **Add Link** button next to the **Child Cue** (Figure 236).

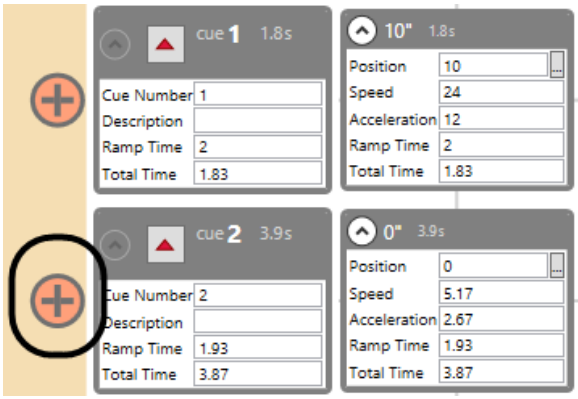


Figure 236

2. Select the **Parent Cue** from the drop down list and select **Time Link**, then press **OK** (Figure 237).
   *In the example pictured below, the selected **Parent Cue** is "Cue #1."*

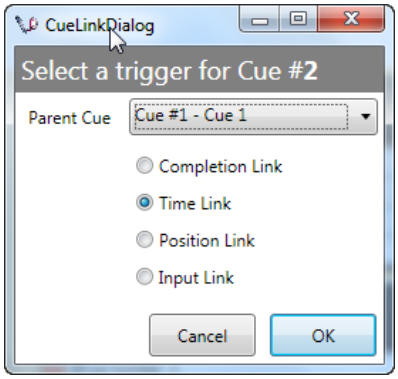


Figure 237

3. The new link is displayed in the left margin. You can edit the **Delay** value to be any number of seconds, including decimal values (Figure 238).
   In the example below, Cue 2 (Child Cue) will not start until Cue 1 (Parent Cue) has run for 10 seconds.

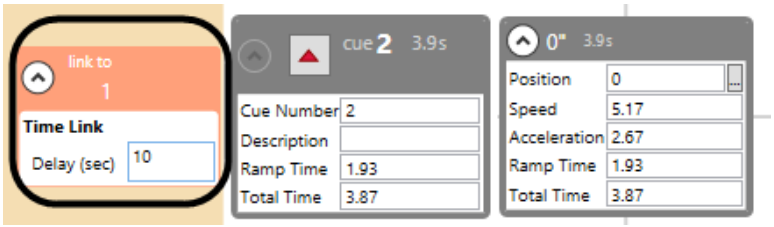


Figure 238

Linking Cues by Motor Position

You can link a **Child Cue** so that it runs when one of the motors in the **Parent Cue** reaches a defined *position.* This is handy when you have to ensure that one motor has cleared the way for another motor.

1. Click the **Add Link** button next to the **Child Cue** (Figure 239).

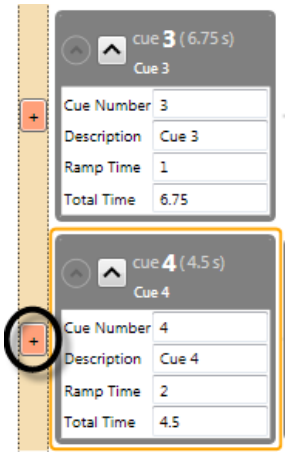


Figure 239

2. Select the **Parent Cue** from the drop-down list and select **Position Link**, then press **OK** (Figure 240).
   *In the example pictured below, the selected **Parent Cue** is "Cue #3."*

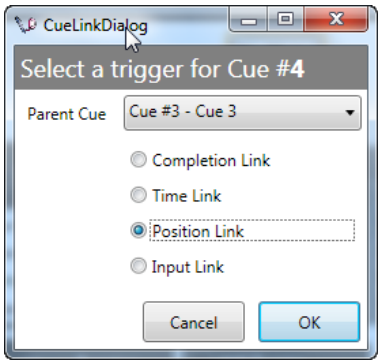


Figure 240

3. The new link will appear in the left margin next to the **Child Cue**. Select the **Motor** from the **Parent Cue** and enter a position that will trigger the link (Figure 241).
   *In the example below, Cue 4 (Child Cue) will not start until the **Small Motor** in Cue 3 (Parent Cue) reaches the 50" mark.*

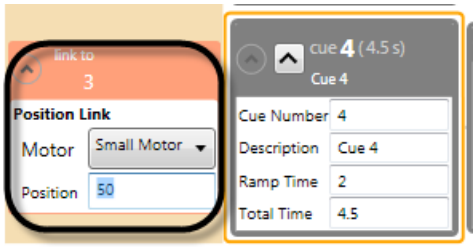


Figure 241

Linking Cues by FX Input

You can link a **Child Cue** to a **Parent Cue** so it runs as soon as an input switch is detected from a Stagehand FX. This can be handy if you have two cues that need to run in close proximity, but require a crew member to give a "Clear" signal for the second cue. In such a case, you could wire a pushbutton to the input of a Stagehand FX. When the **Parent Cue** runs, the crew member would push the button to start the **Child Cue** when appropriate.

1. Click the **Add Link** button next to the **Child Cue** (Figure 242).

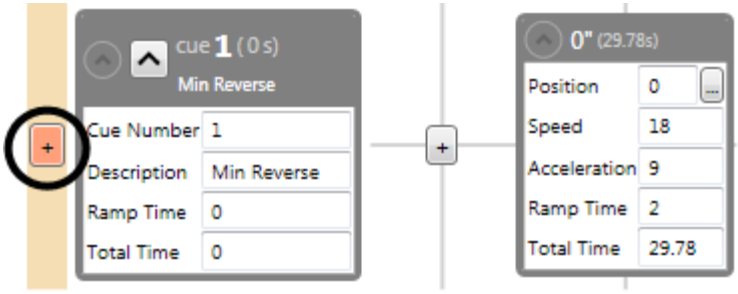


Figure 242

2. Select the **Parent Cue** from the drop-down list and select **Input Link**, then press **OK** (Figure 243).

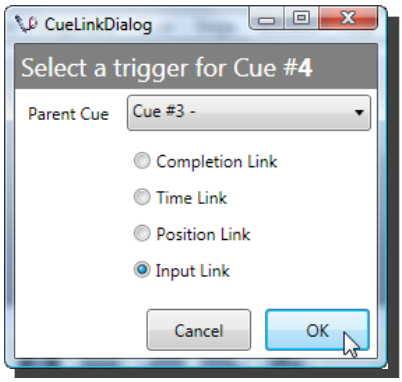


Figure 243

3. The new link will appear in the left margin of the **Cue Grid**. Select the Stagehand FX and the input that should trigger the **Child Cue** (Figure 244).

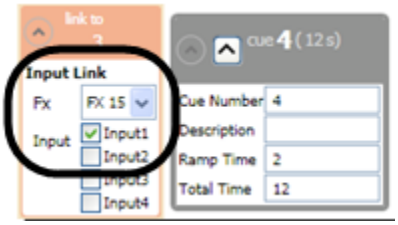


Figure 244

Deleting a Cue Link

1. Right-click on the **Link** in the **Cue Grid** and select **Delete Link** (Figure 245).

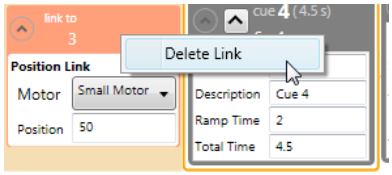


Figure 245

2. Click **Yes** in the **Remove Link** confirmation dialog box (Figure 246).

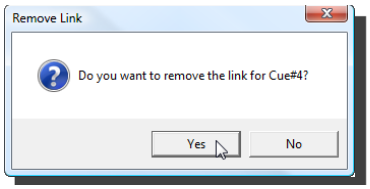


Figure 246

Multi-Speed/Multi-Position Links

There are some cues that require a motor to change *speed* in the middle of the movement. Perhaps a turntable needs to move at a walking pace for 30 seconds and then speed up to running pace and then back down.

Other cues require the motor to oscillate smoothly between two *positions* without coming to a stop and setting the brake, such as a wave which rises and falls smoothly throughout a scene.

Spikemark can handle either of these scenarios with ease. The steps are illustrated in the sections below: "Changing the Speed of a Motor—Manually," "Changing the Speed of a Motor—Automatically" and ""Ping-Pong" a Motor—Changing a Motor's Position."

It should be noted that it is feasible to change *both* the speed and the position of a motor, so that your wave might move between two positions, rising up and down and change speed, moving slowly and then faster.

Changing the Speed of a Motor—Manually (Without a Link)

1. Write two cues for the same motor, with the same target **Position**.
   *In Figure 247, the target **Position** in each cue is "0."*

Figure 247

2. In each Cue, enter the two different **Speed** values.
   *In the example in Figure 248, the **Speed** in Cue 1 is "20" and the **Speed** in Cue 2 is "5." The **Acceleration** is controlled by the first cue, and cannot be altered mid-move.*

Figure 248

3. While Cue 1 is running, load Cue 2 and run it whenever you want the motor to decelerate to the new speed.

Changing the Speed of a Motor—Automatically

1. Write two cues for the same motor, with the same target **Position**.
   *In the example in Figure 249, the target **Position** in each cue is "0."*

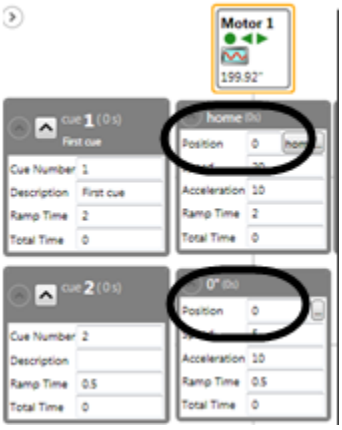


Figure 249

2. In each cue, enter the two different **Speed** values.
   *In the example in Figure 250, the **Speed** in Cue 1 is "20" and the **Speed** in Cue 2 is "5." The **Acceleration** is controlled by the first cue, and cannot be altered mid-move.*

Figure 250

3. Link the two cues together using a **Time Link** or **Position Link**.
   *In the example in Figure 251, Cue 1 and Cue2 are linked with a **Time Link.***

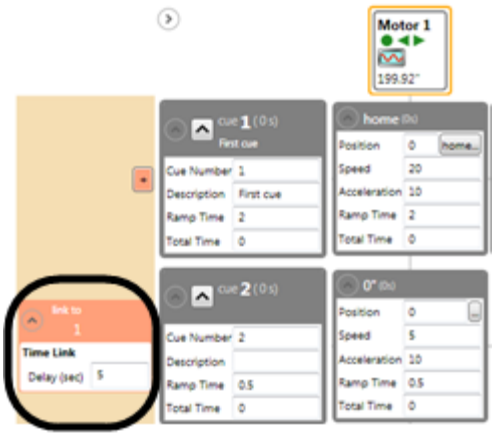


Figure 251

NOTE: When linking two cues to automatically change a motor's speed, do NOT use a Completion Link, as this will bring the motor to a complete stop. Use a Time Link or Position Link in order to smoothly change speeds.

"Ping-Pong" a Motor—Changing a Motor's Position

Creating a "Ping Pong" Motor effect creates an "infinite" loop, where the motor travels between two positions *without coming to a hard stop*. Instead the motor gracefully decelerates to zero-speed, changes direction, and then immediately accelerates in the opposite direction.

Since the “Ping-Pong” affect creates an “infinite” loop, you will have to interrupt the sequence by using any of the following: the **Stop** on the **Cue Controller**; the **Stop All Cues** or **Emergency Stop** buttons on the Showstopper; or a “Soft Stop Motor” Input Action with a Stagehand FX.

1. Write two cues with different **Position** targets for the same motor.
   *In the example in Figure 252, the **Position** targets are “0” and “100.”*

Figure 252

2. Link the two cues together with a **Position Link** or **Time Link**.
   *In the example in Figure 253, Cue 1 links to Cue 2 and Cue 2 links to Cue 1, with **Position Links***

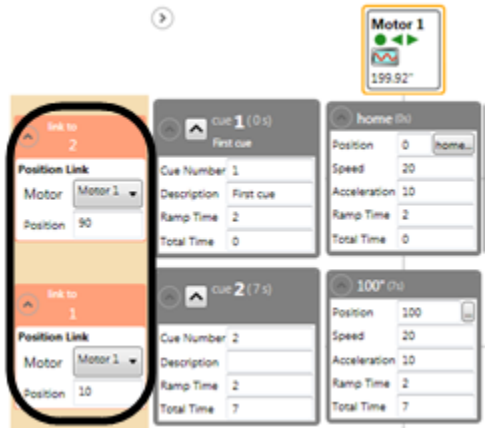


Figure 253

NOTE: When linking two cues to create a "ping-pong” effect, do NOT use a Completion Link, as this will bring the motor to a complete stop. Use a Time Link or Position Link in order to slow the motor to an imperceptible stop, creating a more fluid look for the position change.

The Stage Model Viewer

Spikemark's **Stage Model Viewer** (Figure 254) provides the perfect way to visualize your show. Rather than seeing the bland rows and columns of numbers in the **Cue Grid**, the **Stage Model Viewer** shows you a three-dimensional schematic of your stage. Every motor in the show can be sized and positioned on the virtual stage so it closely resembles the actual scenery in your theatre.

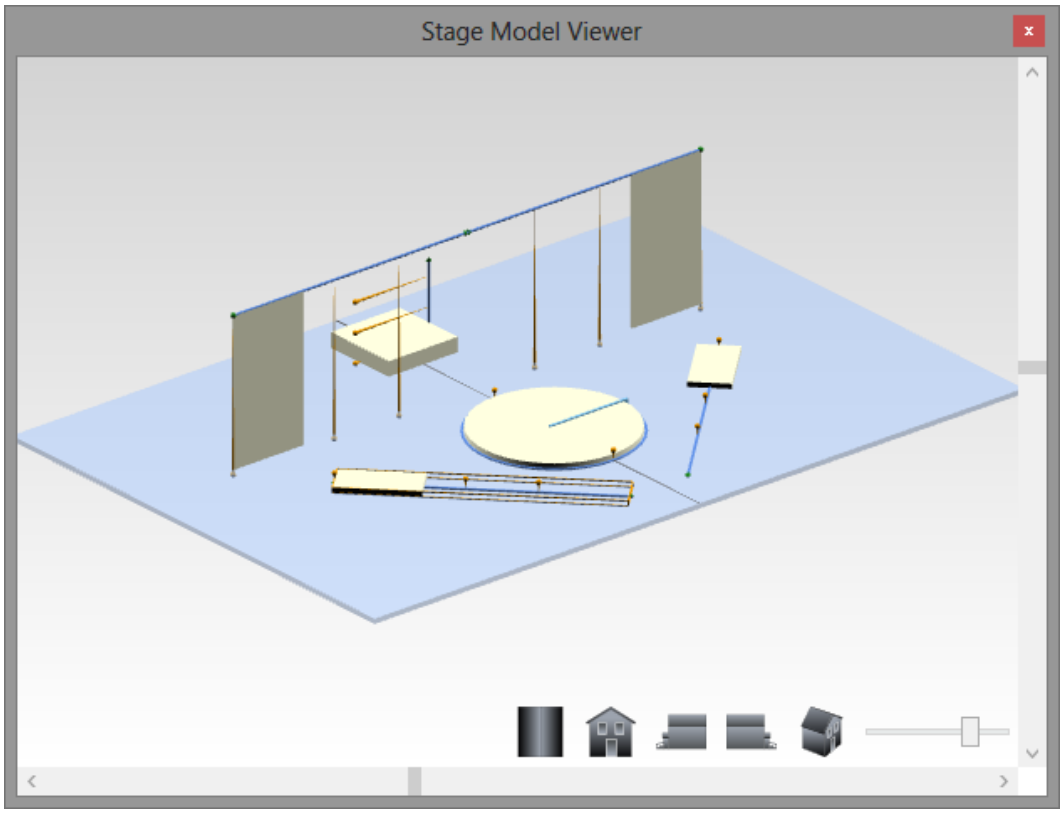


Figure 254

When you load a cue, any motor that is programmed to move will change color and display its destination in the **Stage Model Viewer**, allowing you to see—at a glance—what will move and where it will move to. As the motors move in real life, they move in the **Stage Model Viewer** as well, giving you instant feedback about what is happening on the stage. This feedback is useful when you are sitting back stage trying to keep track of every piece of moving scenery and every person on the stage, which can make it much easier to quickly decide if someone is in harm's way before the Stage Manager shouts, "GO!"

Another great way to use the **Stage Model Viewer** is as a pre-tech cue-writing tool. Using **Spikes** (also see the "Spikes" section on page 163) and the **Stage Model Viewer** you can rough-in your cues and clearly see what the stage will look like before setting foot in the theatre. This can give you and the artistic staff time to play with different looks before tech and speed up the

rehearsal process. By using the Stage Model View and Simulator mode, you can test all of your automation cues without being connected to any machinery. For more information on how to use the **Stage Model Viewer** when you are running shows, see the “Running Shows” section on page 174.

NOTE: To open the Stage Model Viewer in a separate window, click Stage Model Viewer in the Window menu.

Navigating the Stage Model Viewer

You can use your right mouse button to pan around the **Stage Model Viewer** space by clicking and dragging inside the 3D model. You can also use the on-screen tools—listed below—to zoom, pan, and twirl the 3D model.

The **Zoom Slider** (Figure 255) allows you to magnify the **Stage Model Viewer**. You can also use the scroll-wheel on your mouse to zoom in or out.

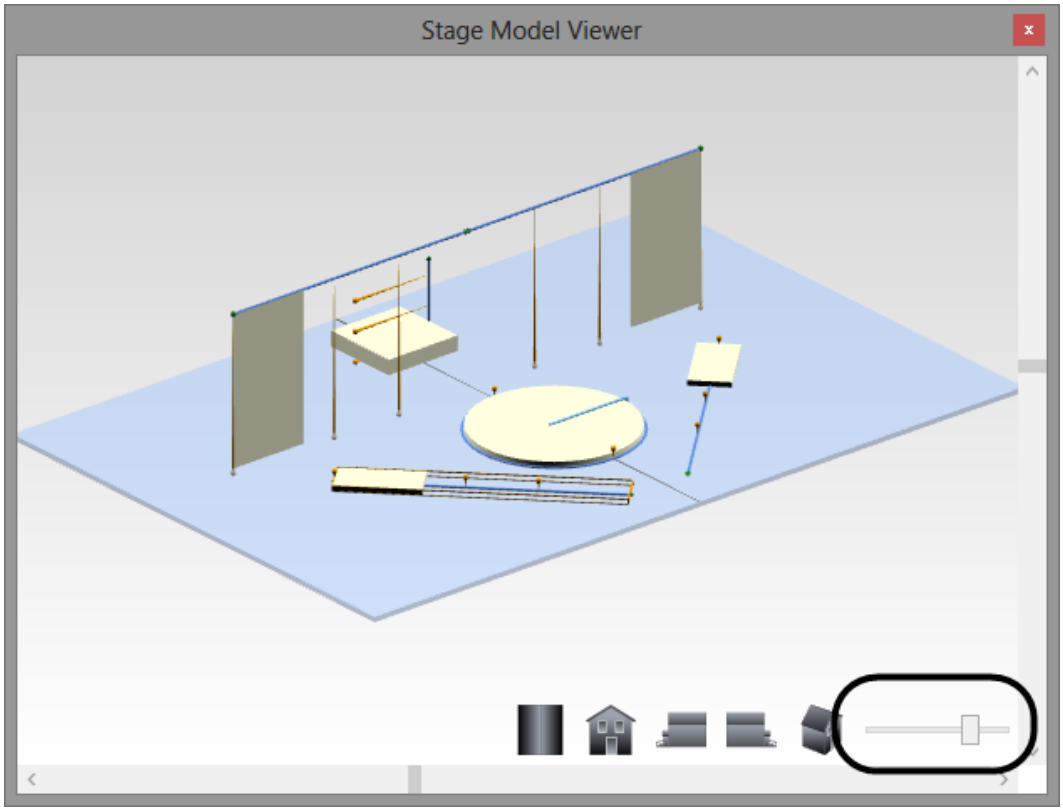


Figure 255

Use the **Scrollbars** (Figure 256) on the right and bottom of the **Stage Model Viewer** to orbit around the stage.

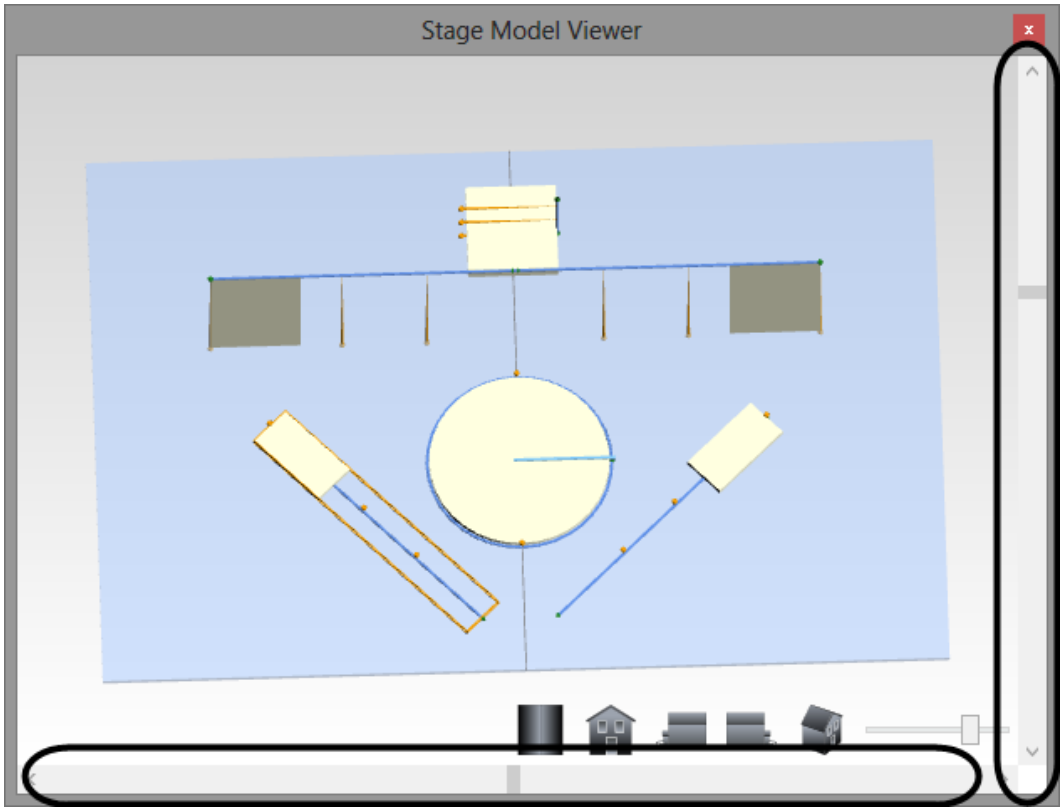


Figure 256

The **Preset View Buttons** allow you to quickly jump to the 5 orthographic projections (Figure 257 through Figure 261).

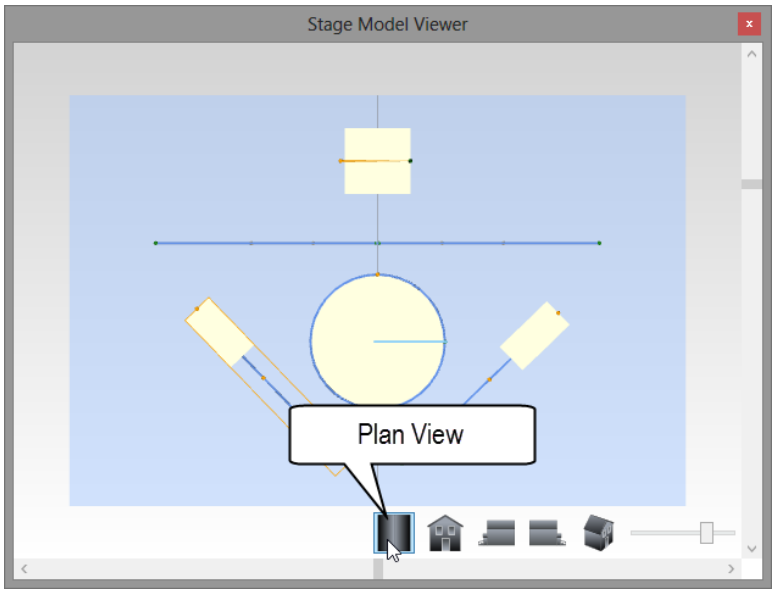


Figure 257

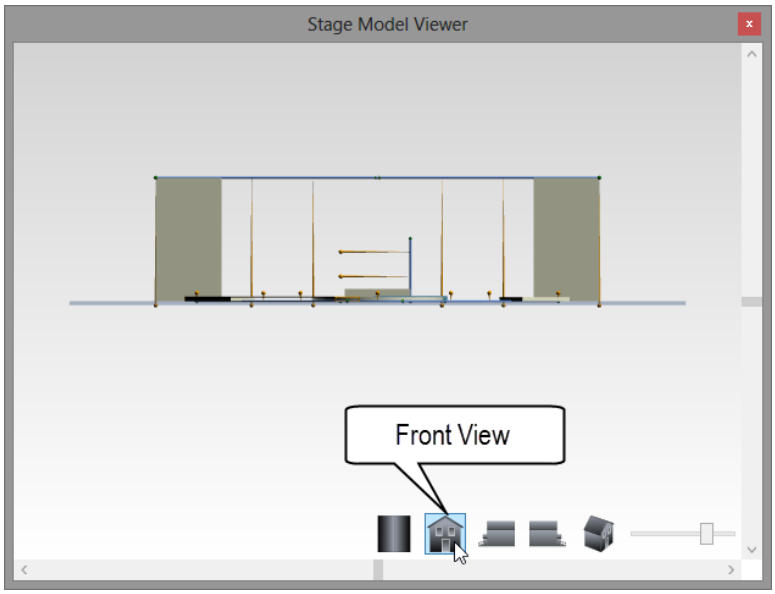


Figure 258

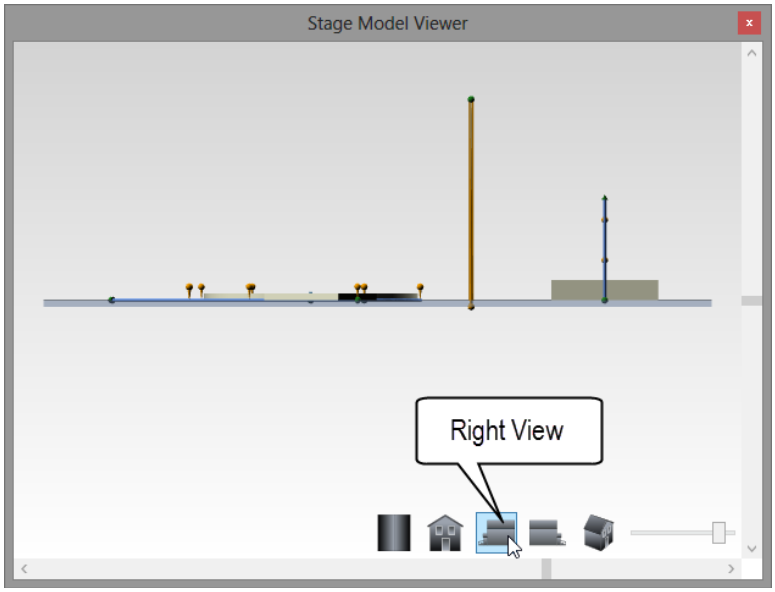


Figure 259

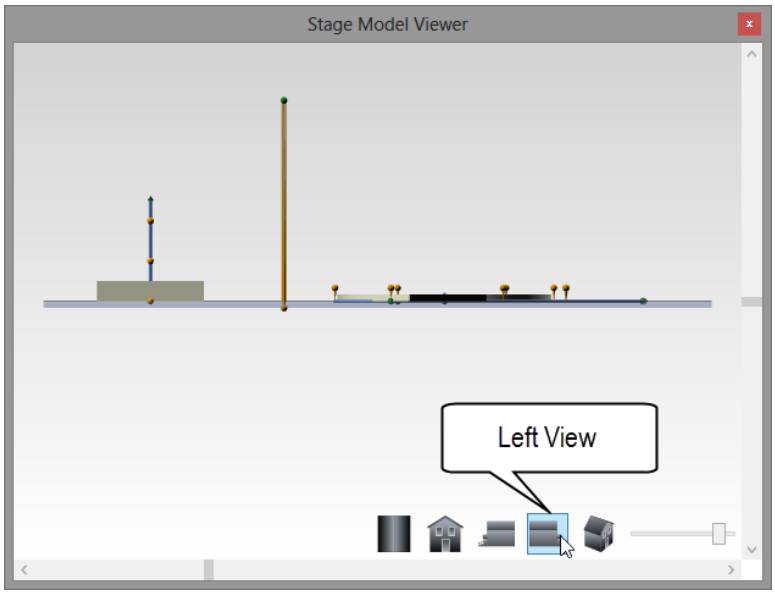


Figure 260

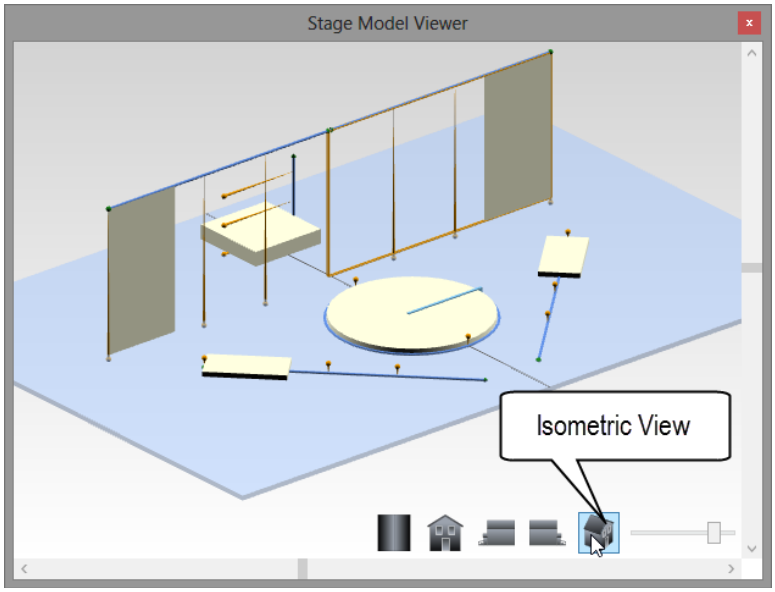


Figure 261

Setting the Stage Dimensions

In the **Stage Model Viewer**, the stage is shown as a semi-transparent box. To make the stage in the **Stage Model Viewer** more closely resemble the size of your stage, you can change the default stage dimensions.

1. From the **Stage** menu, click **Stage Dimensions** (Figure 262).

Figure 262

2. Edit the dimensions of the stage to match your physical stage and click **OK** (Figure 263). *All Stage Dimensions are in feet.*

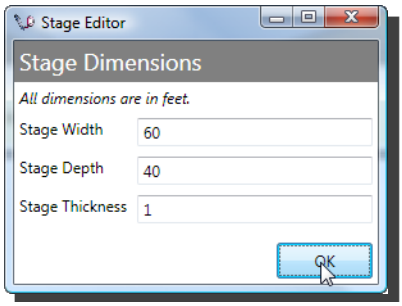


Figure 263

Making a Winch Model

After you've changed the size of the stage to match your theatre space, the next step is to make the motors look roughly like the scenery on the stage. Each motor in the show can be viewed as either a winch or a turntable. Winches always look like a box traveling along a track. The length of the track is determined by the **Max Forward** and **Min Reverse** parameters (set in the **Position** section of the **Stagehand Editing Pane**). The size and placement of the scenery on the track can be adjusted in the **Schematic** section of the **Stagehand Editing Pane**.

1. Select the motor you wish to adjust in the **Cue Grid.**

2. In the **Stagehand Editing Pane,** scroll to the **Schematic** section and make sure that **Winch** is selected at the bottom (Figure 264).
Adding a Turntable is described in the "Making a Turntable Model" section on page 159.

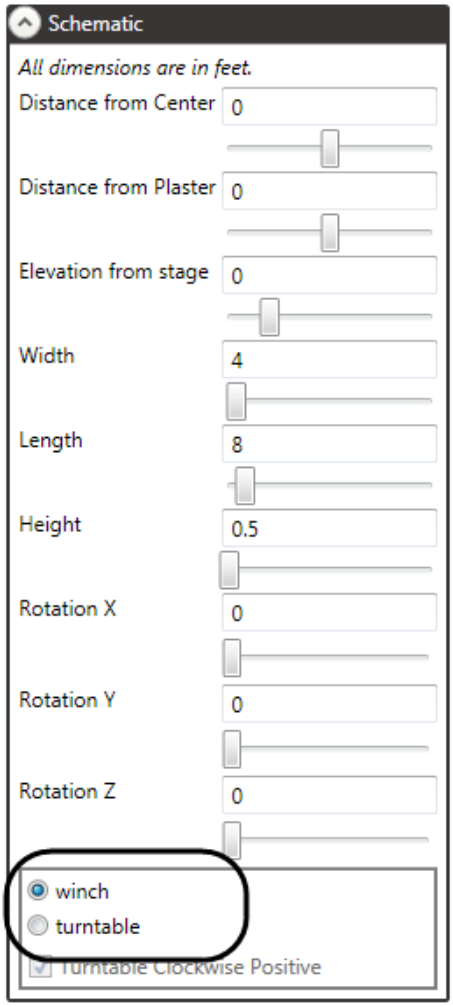


Figure 264

3. Adjust the **Width**, **Length**, and **Height** properties (Figure 265) by using either the sliders beneath each text field or typing in a dimension directly. As you change the dimensions, the winch model in the **Stage Model Viewer** will update.
   Note that all dimensions are given in feet.

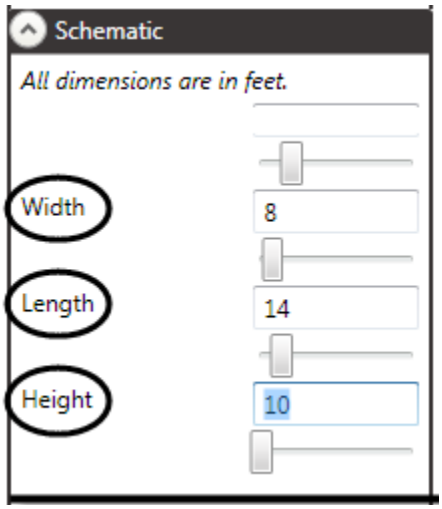


Figure 265

4. Now, move the winch into position so that it matches your stage. Use the **Distance from Center**, **Distance from Plaster** and **Elevation from stage** field sliders (Figure 266) to get the position close. You can then fine tune the position by typing in accurate dimensions from your ground plan.

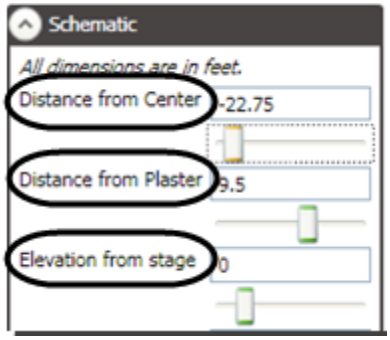


Figure 266

NOTE: By default, a winch is displayed with its minimum reverse position Stage Right [the Min Rev Position **is set in the Position section of the Stagehand Editing Pane] and its forward direction heading Stage Left. You can use the** Rotation Y **parameter [in the Schematic section of the Stagehand Editing Pane] to spin the winch around and reposition it if you need the** Min Rev Position **to be Stage Left and its forward directing heading Stage Right. (See Figure 267, Figure 268 and Figure 269).**

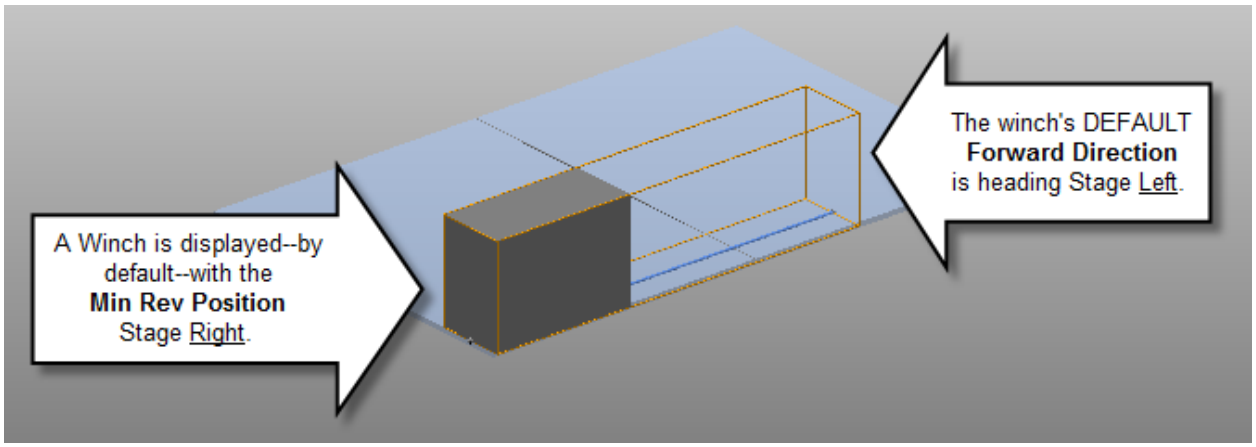

A Winch is displayed--by default--with the **Min Rev Position** Stage Right.
The winch's DEFAULT **Forward Direction** is heading Stage Left.

Figure 267

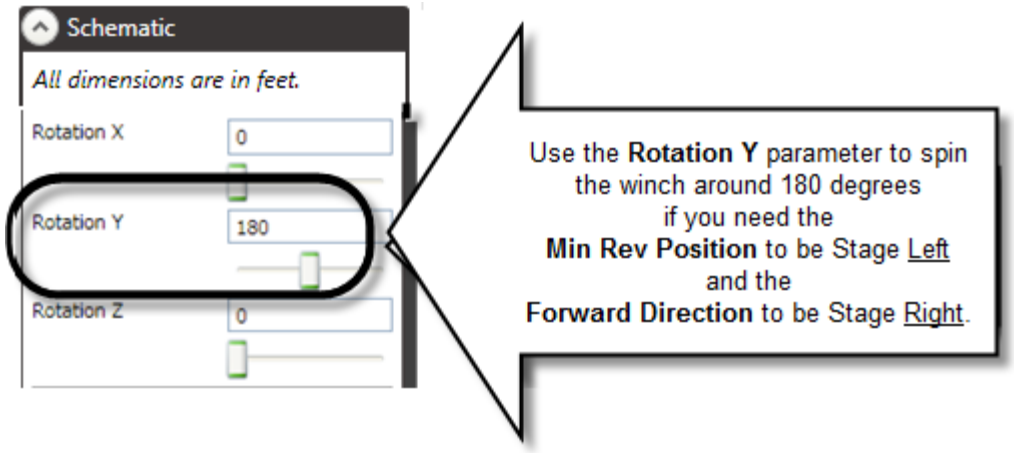

Schematic
All dimensions are in feet.
Rotation X
0
Rotation Y
180
Rotation Z
0
Use the **Rotation Y** parameter to spin the winch around 180 degrees if you need the **Min Rev Position** to be Stage Left and the **Forward Direction** to be Stage Right.

Figure 268

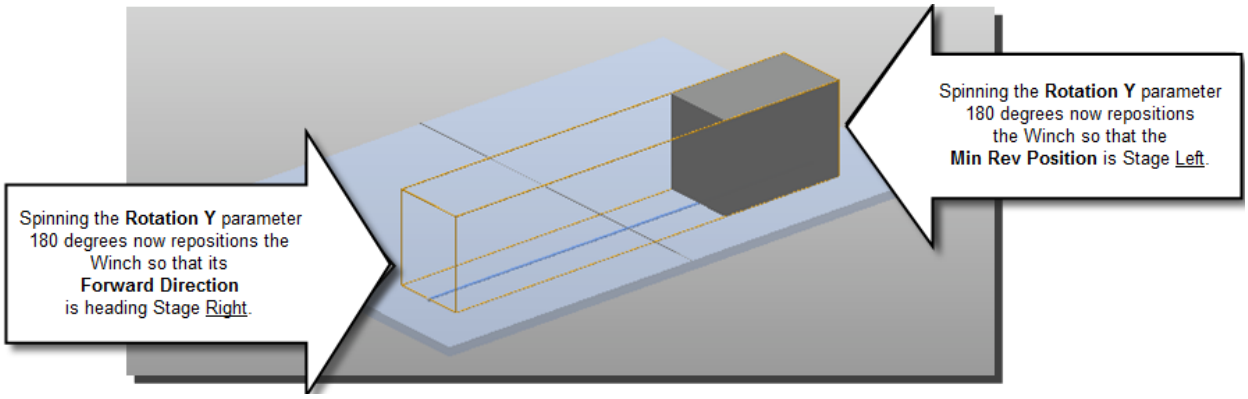

Spinning the **Rotation Y** parameter 180 degrees now repositions the Winch so that its **Forward Direction** is heading Stage Right.
Spinning the **Rotation Y** parameter 180 degrees now repositions the Winch so that the **Min Rev Position** is Stage Left.

Figure 269

Making a Lift Model

The winch-style model can be used to represent various machines other than ordinary deck winches. By adjusting the size and rotation of the winch we can easily make it look like a Lift.

1. Select the motor you wish to adjust in the **Cue Grid**.

2. In the **Stagehand Editing Pane**, scroll until you find the **Schematic** section and make sure that **Winch** is selected at the bottom (Figure 270).
   Adding a Turntable is described in the "Making a Turntable Model" section on page 159.

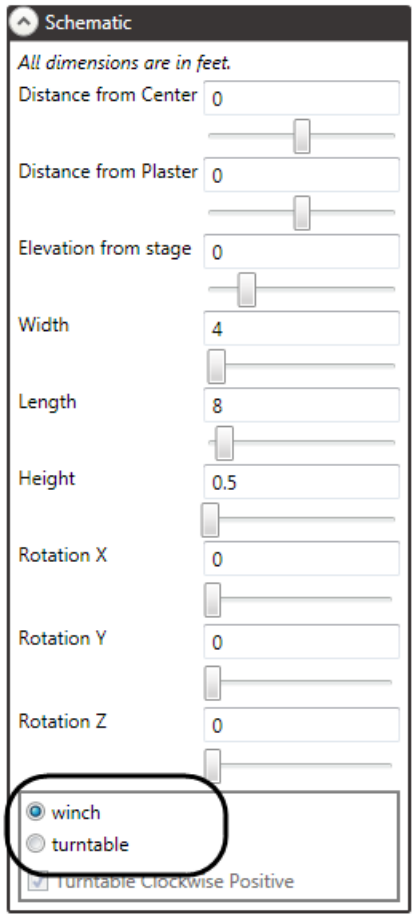


Figure 270

3. Using the **Rotation X** and **Rotation Z** sliders, rotate the winch model until it is standing upright (Figure 271).

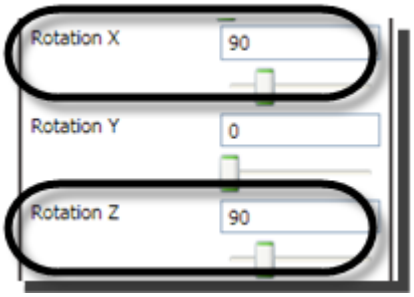


Figure 271

4. Now use the **Width**, **Length**, and **Height** sliders to make the winch look like a lift platform (Figure 272 and Figure 273).

Figure 272

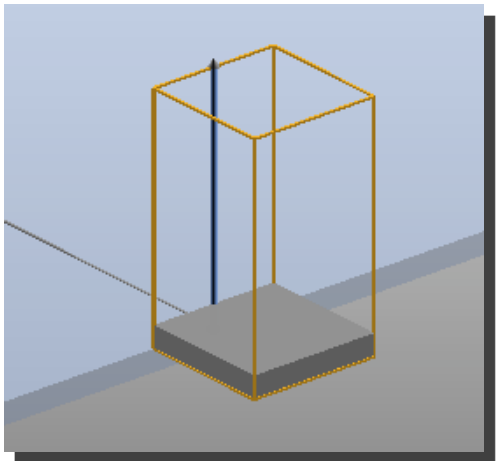


Figure 273

5. Using the **Elevation from stage** field, position the Lift so that it travels into the trap room (Figure 274 and Figure 275).
*Use the **Distance from Center** and **Distance from Plaster** fields to place your Lift in the appropriate part of your stage.*

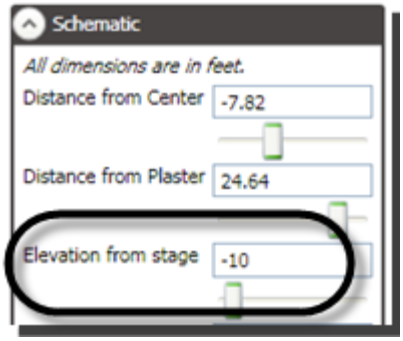


Figure 274

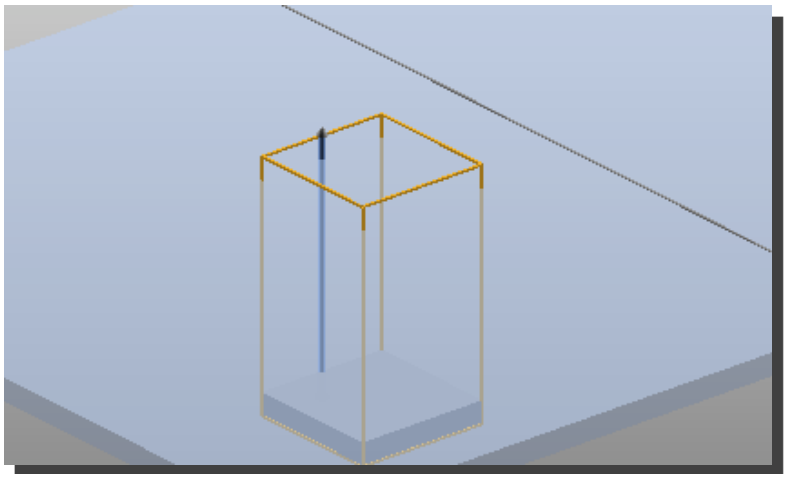


Figure 275

NOTE: The stage is semi-transparent so that you can see scenery moving, even when it is beneath the stage.

Making a Curtain Model

The winch-style model can be used to represent various machines other than ordinary deck winches. By adjusting the size and rotation of the winch, we can easily make it look like either a traveling curtain or a flying curtain.

1. Select the motor you wish to adjust in the **Cue Grid**.
2. Scroll to the **Schematic** section of the **Stagehand Editing Pane** and make sure that winch is selected at the bottom of the list (Figure 276).
   Adding a Turntable is described in the Making a Turntable Model section on page 159.

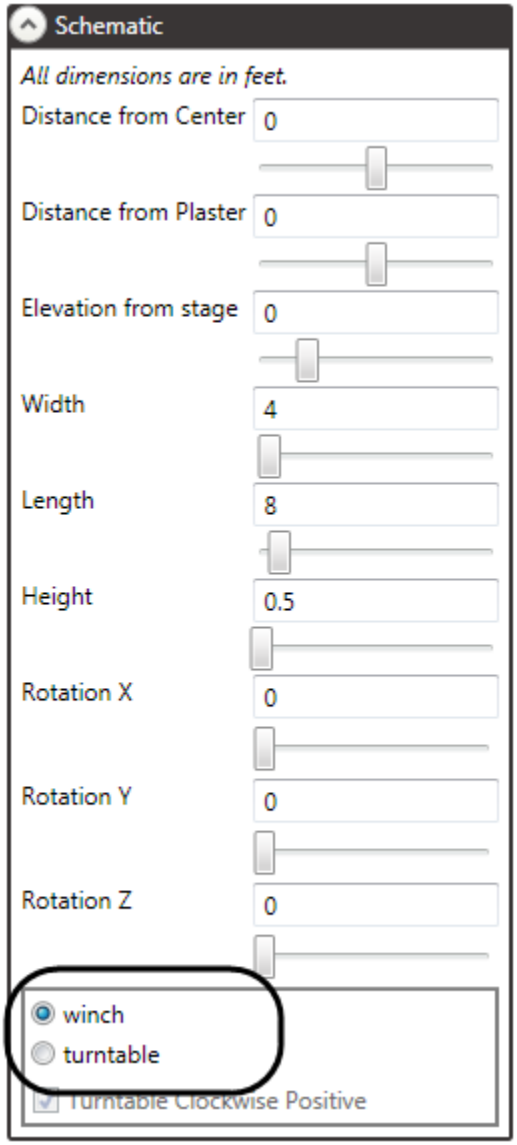


Figure 276

3. Adjust the **Width**, **Length**, and **Height** properties to make a tall, thin curtain (Figure 277 and Figure 278).

Figure 277

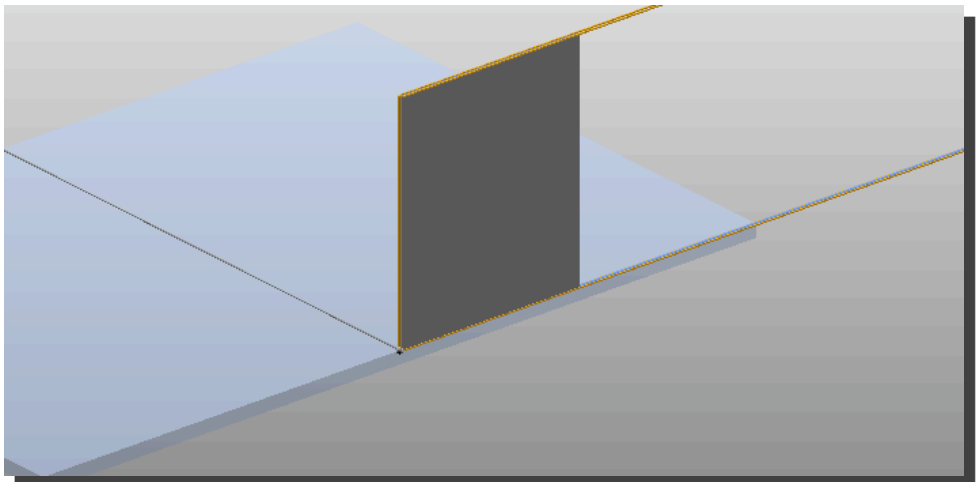


Figure 278

4. Using the **Distance from Center**, **Distance from Plaster** and **Elevation from Stage** fields, position the curtain on the stage (Figure 279 and Figure 280).

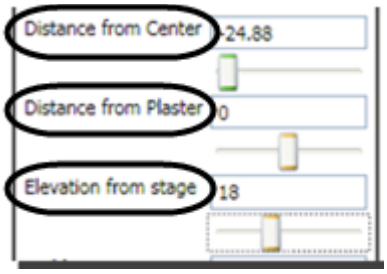


Figure 279

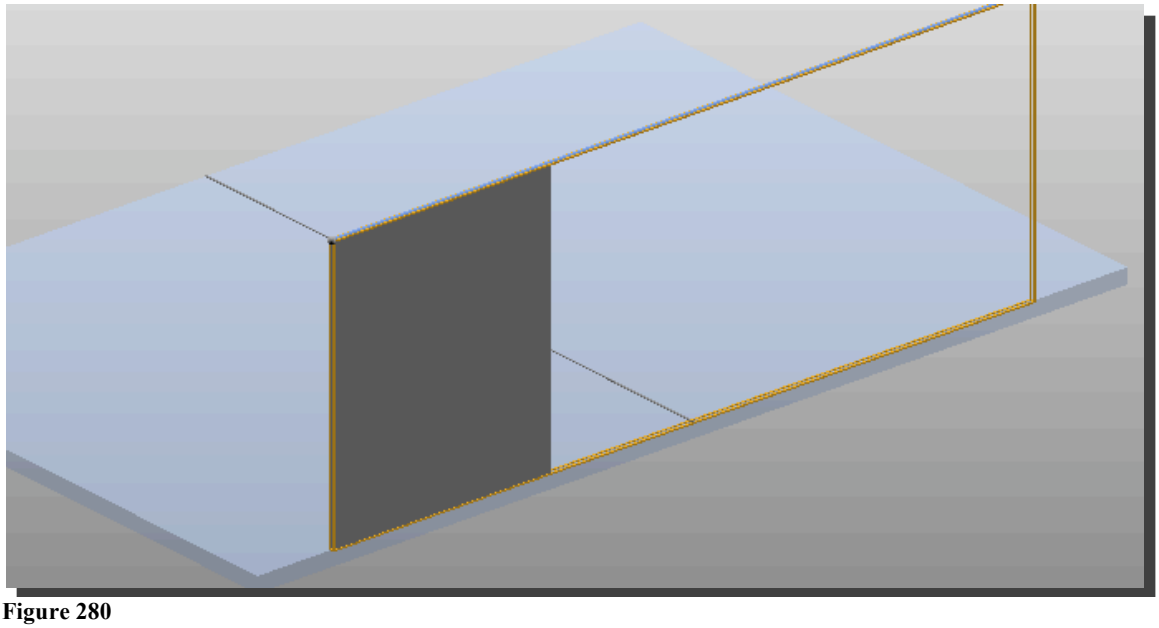


Figure 280

5. By adjusting the **Rotation Z**, **Distance from Center**, **Distance from Plaster**, **Length** and **Height** fields, you can change the horizontal traveler curtain into a vertical flying curtain (Figure 281 and Figure 282).

Figure 281

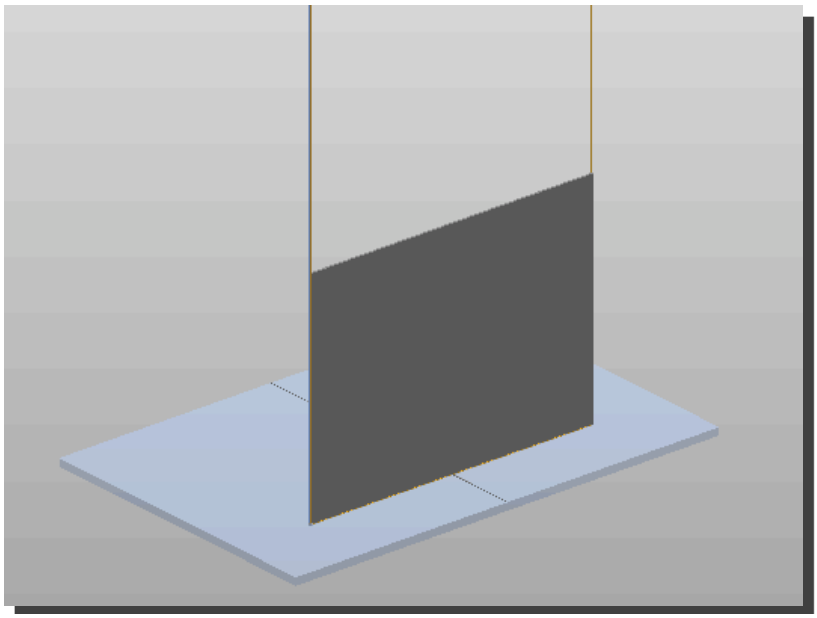


Figure 282

Making a Turntable Model

The default model style for every motor in the show is a winch, which moves linearly along a track. This works great for deck winches, lifts, and curtains which all move in a linear fashion.

However, a lot of shows include turntables. Spikemark makes it easy to change a motor model from winch to turntable.

1. Select the motor you wish to adjust in the **Cue Grid**.
2. Scroll to the **Schematic** section of the **Stagehand Editing Pane** and select **Turntable** at the bottom of the screen (Figure 283).

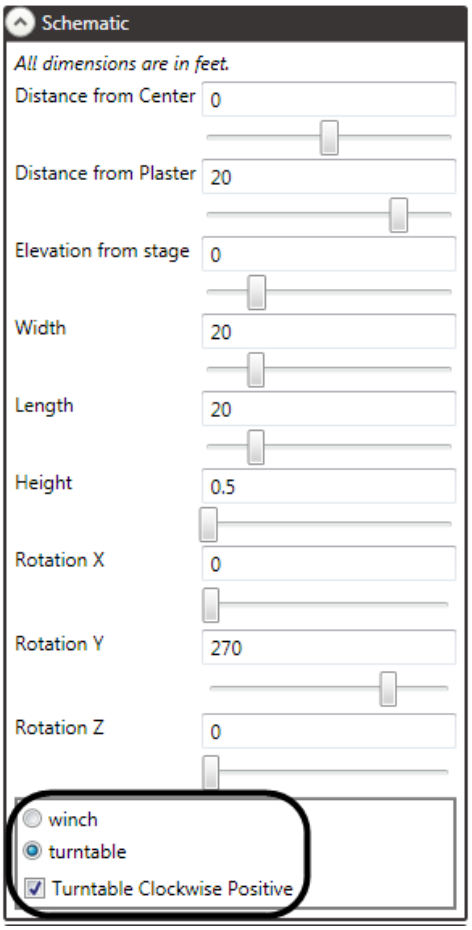


Figure 283

3. Use the **Distance from Plaster** and the **Width** sliders to position and resize the turntable (Figure 284 and Figure 285).

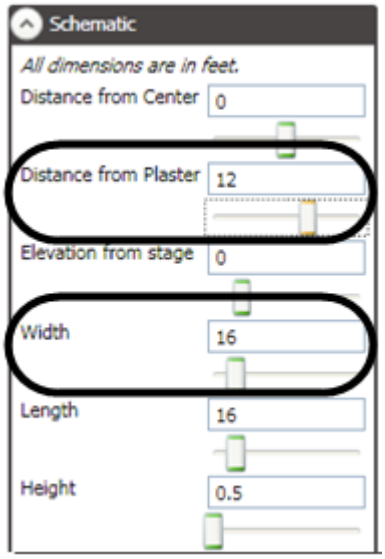


Figure 284

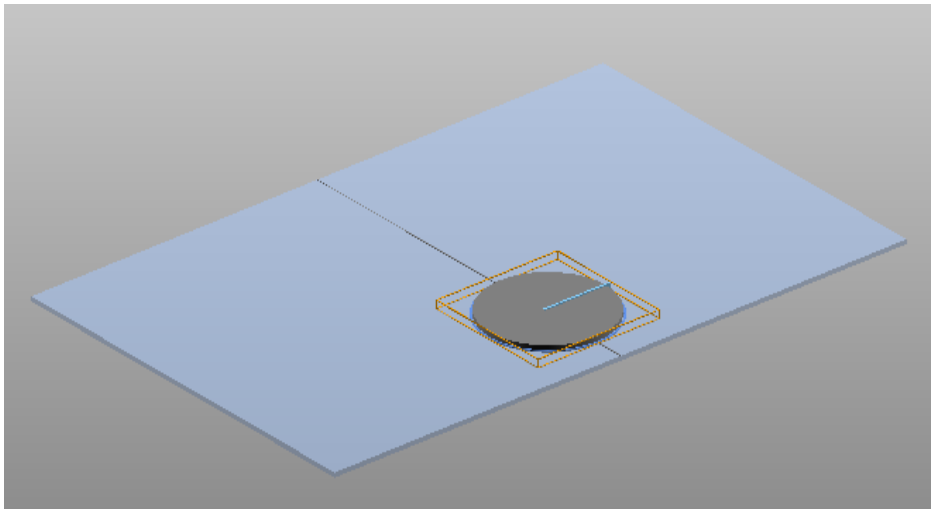


Figure 285

NOTE: By default, the turntable model orients position zero ("0 degrees") towards Stage Left. You can change this orientation by changing the Rotation Y parameter (see Figure 286, Figure 287 and Figure 288).

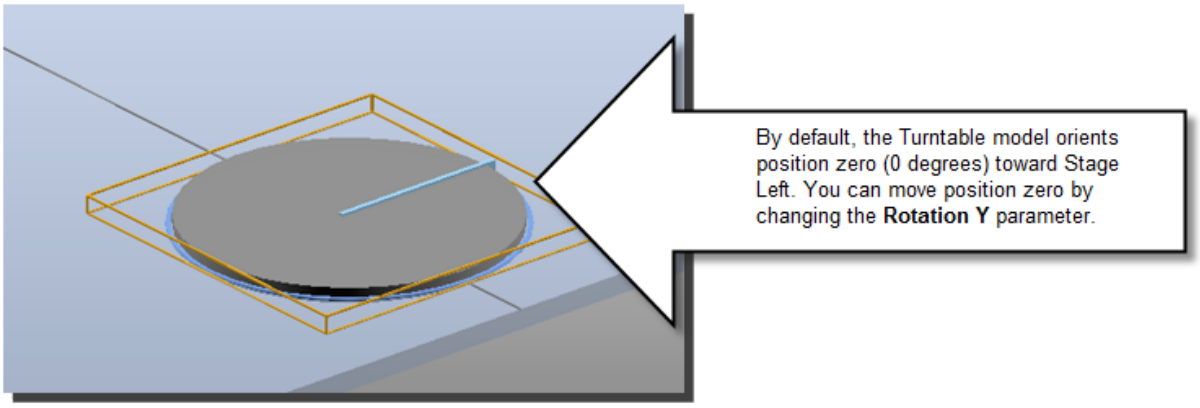

By default, the Turntable model orients position zero (0 degrees) toward Stage Left. You can move position zero by changing the **Rotation Y** parameter.

Figure 286

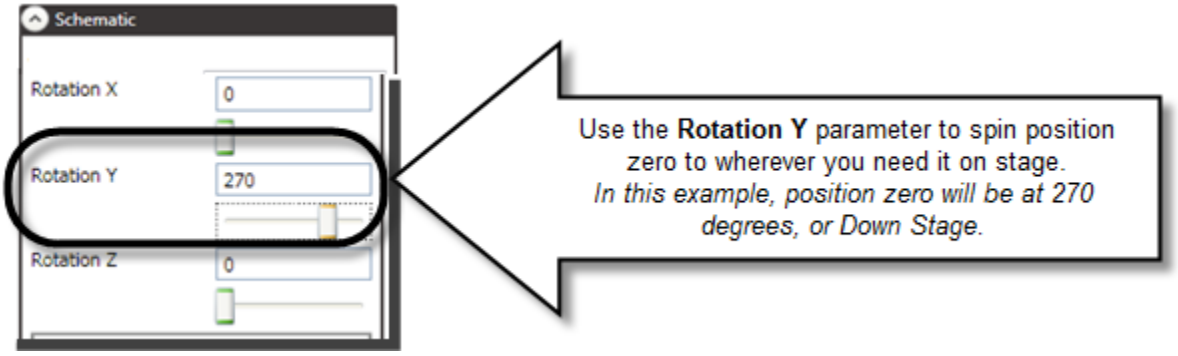

Schematic
Rotation X
0
Rotation Y
270
Rotation Z
0
Use the **Rotation Y** parameter to spin position zero to wherever you need it on stage.
In this example, position zero will be at 270 degrees, or Down Stage.

Figure 287

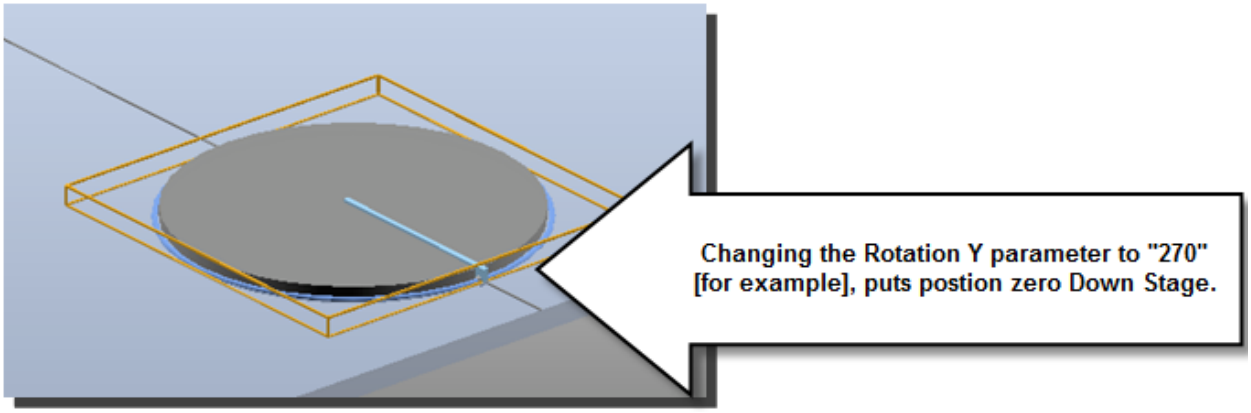


Figure 288

Spikes

The **Spikes** feature is designed to mimic the usefulness of real spike tape. Rather than trying to remember bizarre motor position numbers (like 251.13”), you can now assign names to commonly used positions (‘on stage’ = 251.13”).

Often during the rehearsal process, the **Spike** positions need to be tweaked. If you were physically pushing the wagons around, you would just move the spike tape on the stage floor to a new position. Similarly, with the new **Spikes** feature, you simply edit the position of the **Spike** in one place and all cues that contain a motor that uses the **Spike** are automatically updated, saving you from manually having to make the change throughout all the cues where that motor is used.

Adding Spikes within Cues—Using the Spikes Position Button

1. Click on the applicable Cue in the **Cue Grid** that contains the motors you wish to set the Spikes for (Figure 289).

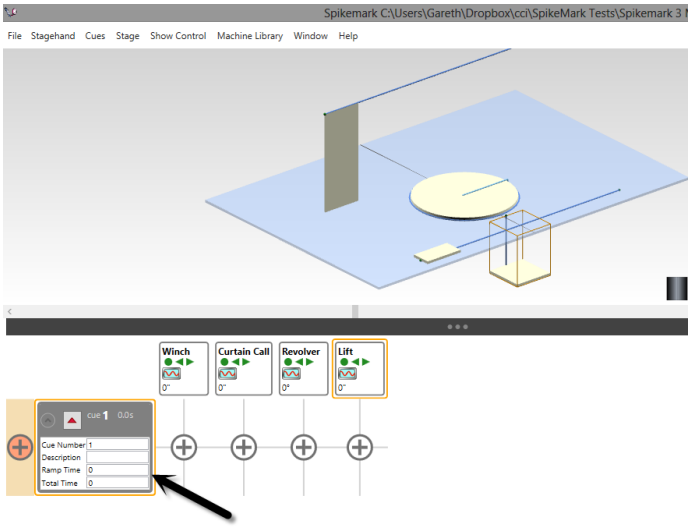


Figure 289

2. Select the motor to which you wish to add a **Spike** (Figure 290).

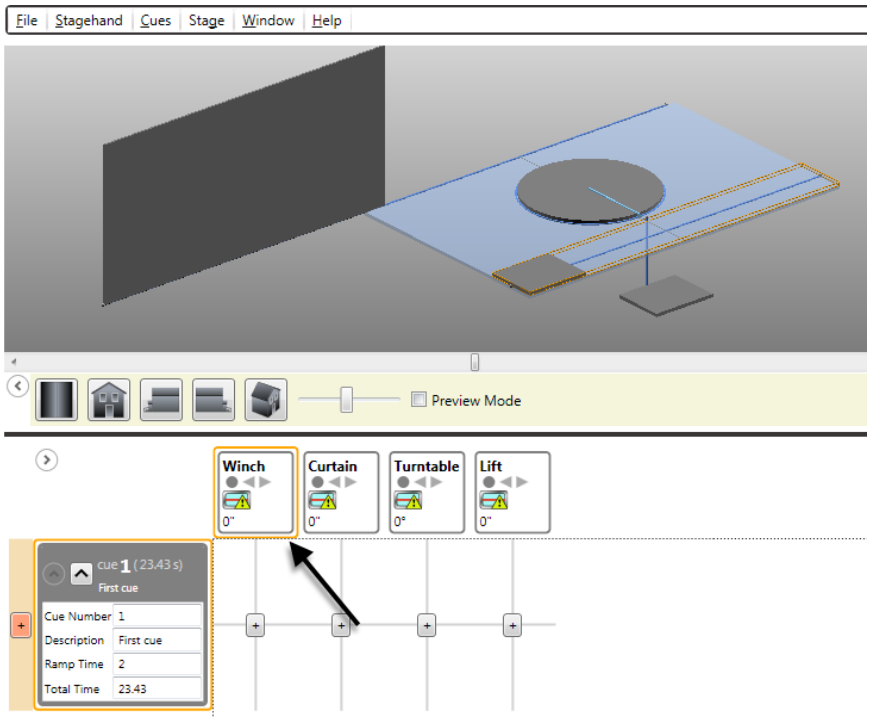


Figure 290

3. If you don't already have a movement for the selected motor in this cue, click the **Add Movement** button in the **Cue Grid** (Figure 291).

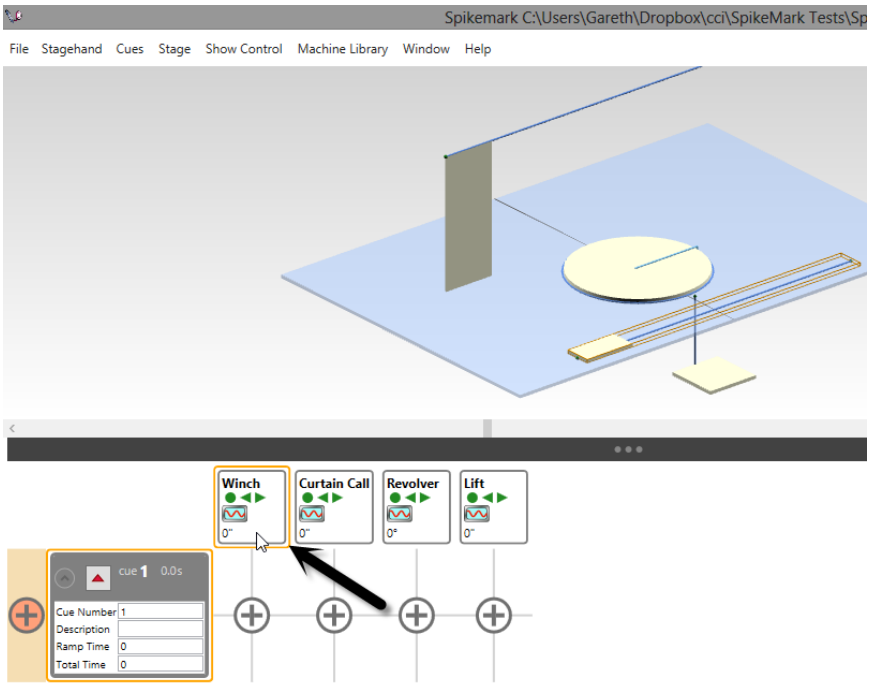


Figure 291

4. Click on the **Spike Position** button in the motor's movement window (Figure 292).

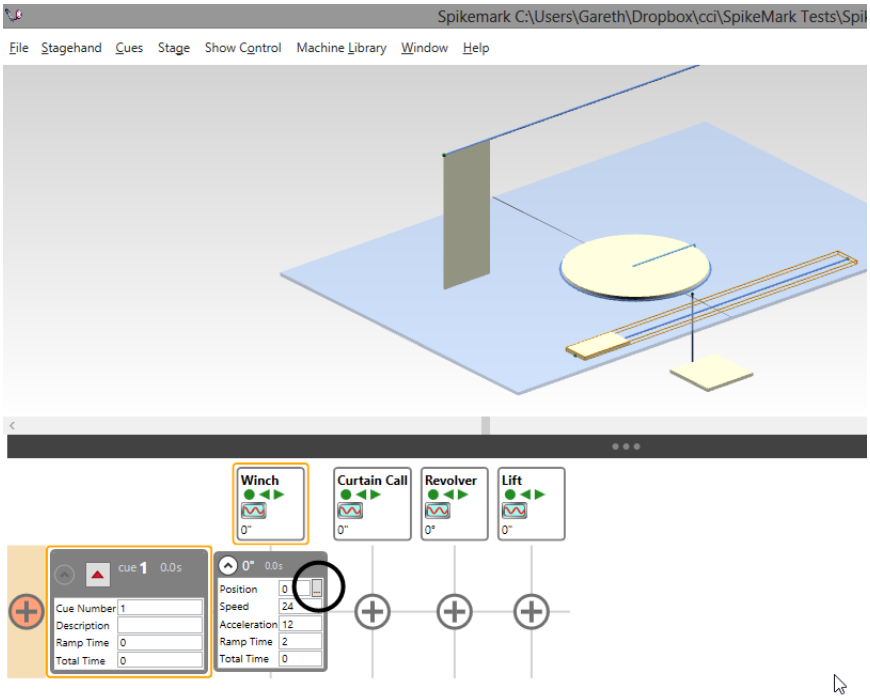


Figure 292

5. Click on the **Add Spike** button in the Spikes window (see Figure below).

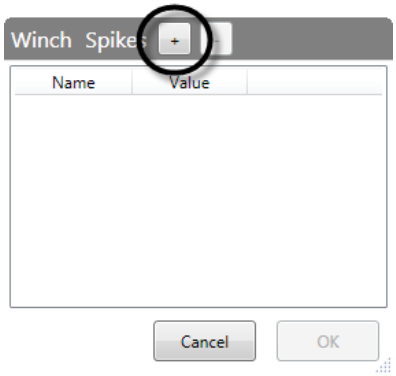


Figure 293

6. A new Spike will be added with the generic **Name** of "spike" and a **Value** of "0" (Figure 294).

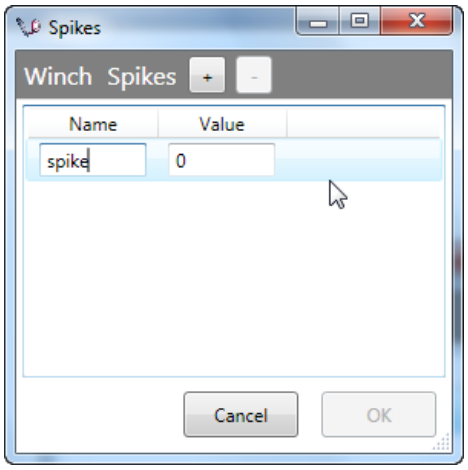


Figure 294

7. Enter a meaningful **Name** and the Spike position **Value**. Select the Spike and click **OK** (Figure 295).

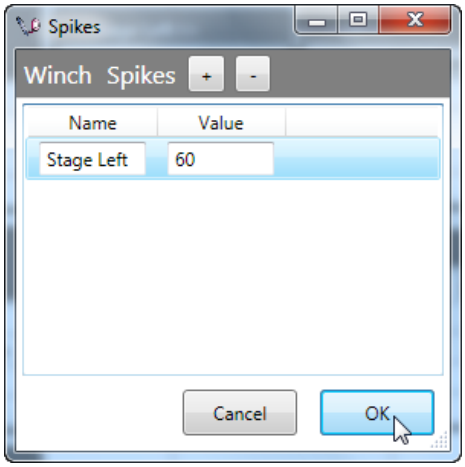


Figure 295

8. The selected Spike will display in the motor's movement for the cue (Figure 296). *In the example below, the Winch will move to the defined Spike—"Stage Left 60"—in Cue 1.*

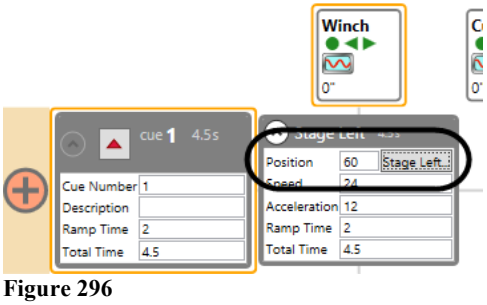


Figure 296

Adding Spikes within Cues—Using the Stagehand Editing Pane

1. In the **Cue Grid**, select the motor to which you wish to add Spikes (Figure 297).

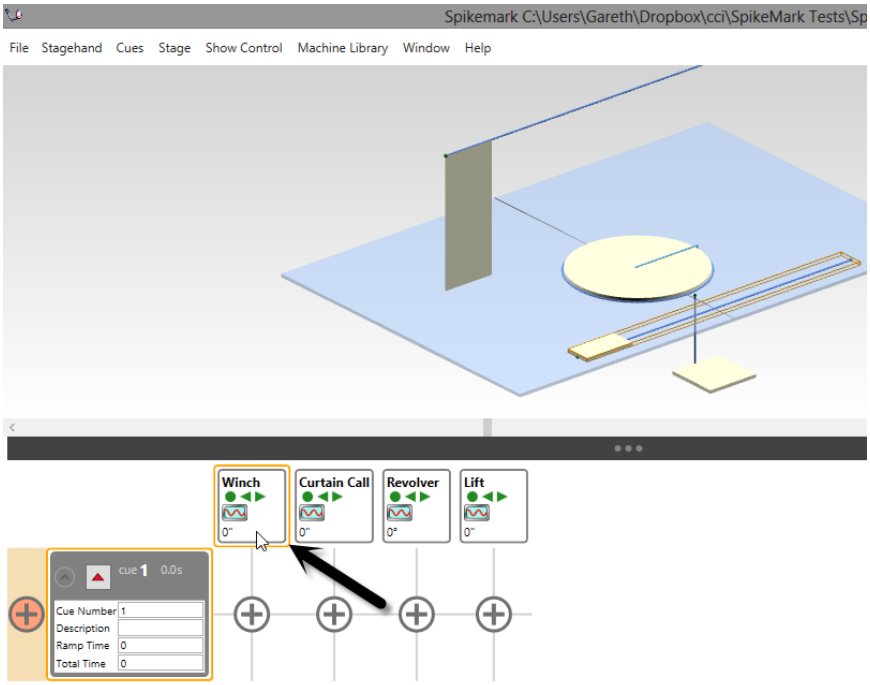


Figure 297

2. Scroll to the **Spikes** section in the **Stagehand Editing Pane** and click **Add Spike** (Figure 298).

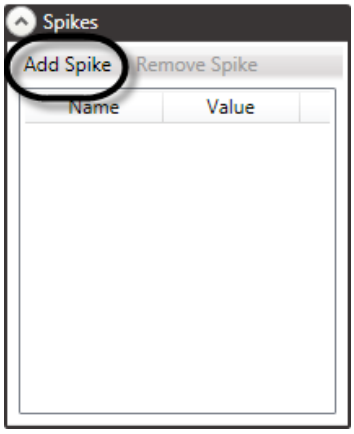


Figure 298

3. Enter a logical **Name** and the **Value** (the measurement where the Spike should be located).

*In the example in Figure 299, the **Name** of the Spike is "Stage Right" and the **Value,** or where the Spike will be placed, is position"0."*

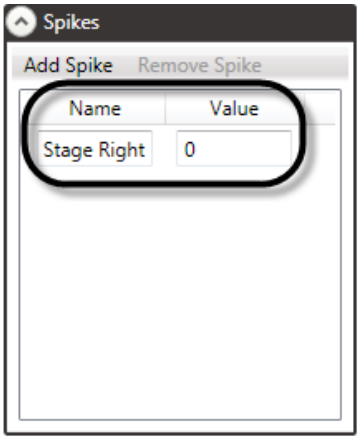


Figure 299

4. You can repeat steps 2 and 3 until you have all the Spikes for this motor.
In Figure 300, the Spikes "Stage Left" and "Center" have also been added.

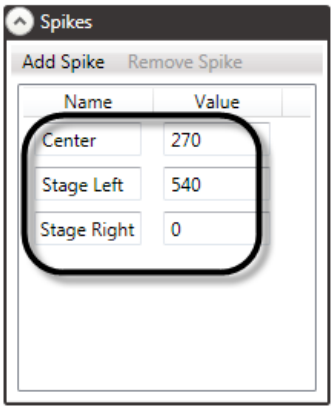


Figure 300

5. You can now set the position for the motors within any given cue by clicking on the applicable Cue in the **Cue Grid** (Figure 301).

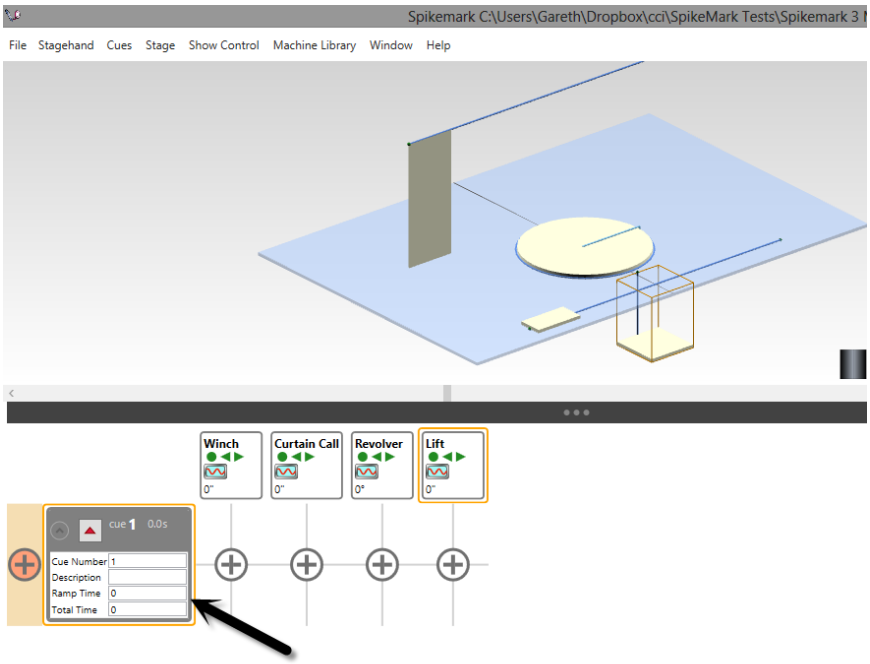


Figure 301

6. If you don't already have a movement for the selected motor in this cue, click the **Add Movement** button in the **Cue Grid** (Figure 302).

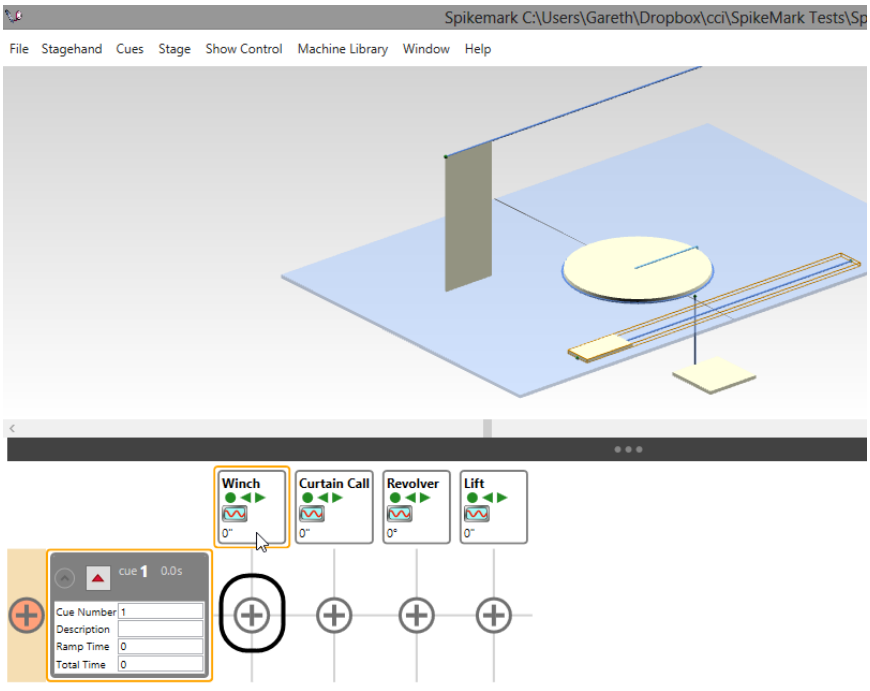


Figure 302

7. Click on the **Spike Position** button in the motor's movement window (Figure 303).

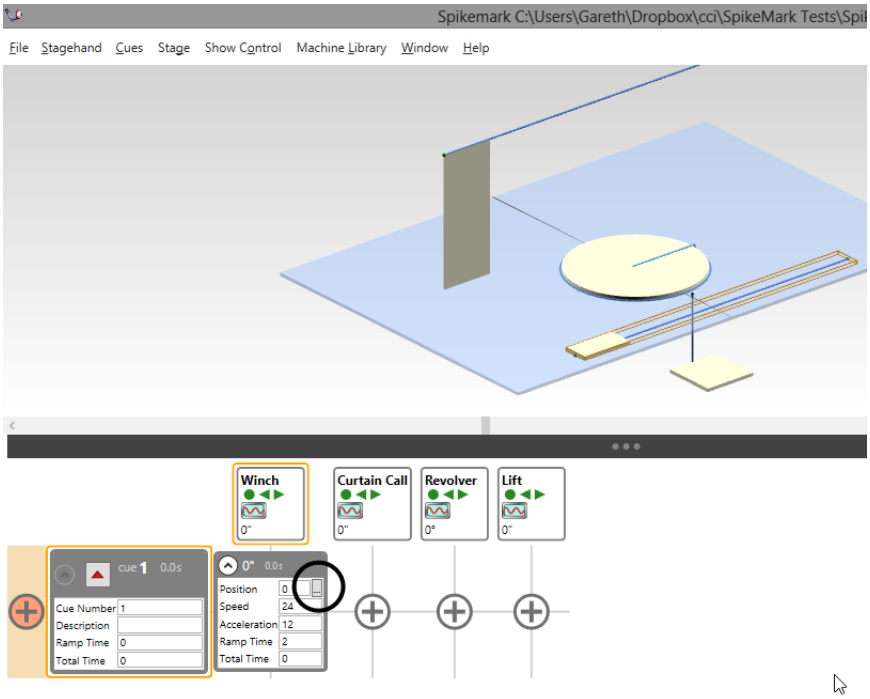


Figure 303

8. Select the Spike where you want this motor to move in this cue and click **OK**.
In Figure 304, the Spike "Center 270" has been selected.

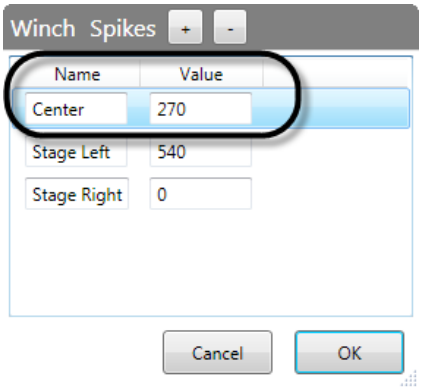


Figure 304

9. The selected Spike will display in the motor's movement for the cue (Figure 305).
In the example below, the Winch will move to the defined Spike—Center, 270—in Cue 1.

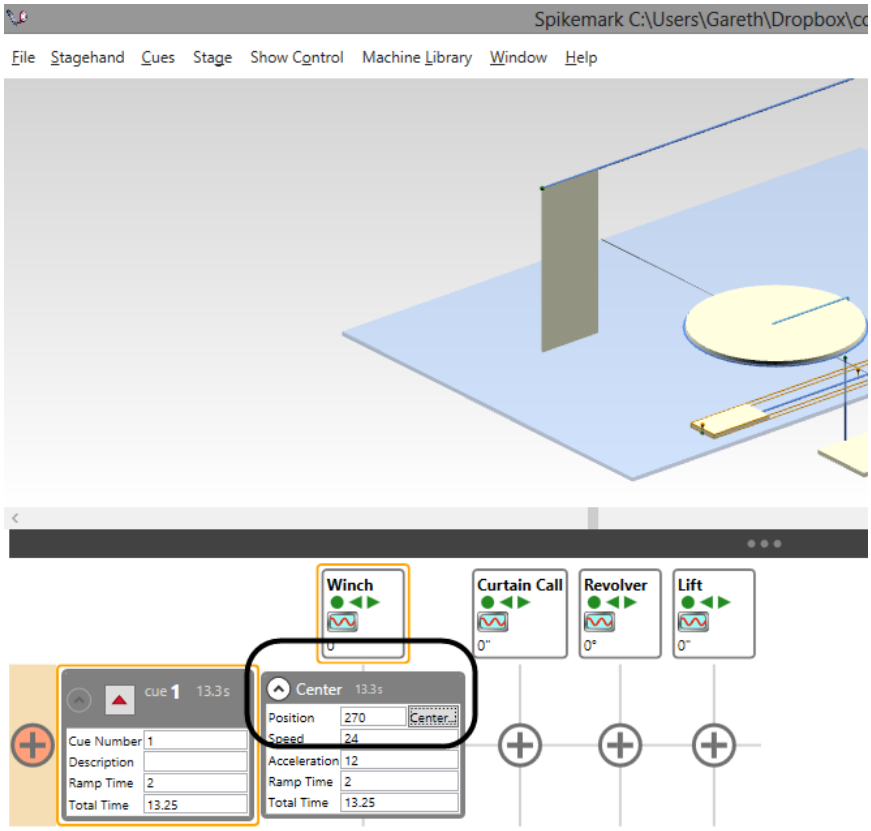


Figure 305

Changing a Spike Position

A key advantage to using **Spikes** is that you can edit the spike value and all cues that reference the spike will be updated. This makes it very easy to tweak position values and avoid tracking the changes manually through the entire show. For example, we've defined a spike named "midstage." However, during the rehearsal process it becomes evident that the "midstage" position needs to move onstage a quarter of an inch. We've already used this spike in a bunch of cues, so to update the spike and have the new position updated in all cues we just edit the spike.

1. In the **Cue Grid**, select the motor that has the Spike you wish to edit (Figure 306).

Figure 306

2. Go to the **Spikes** section of the **Stagehand Editing Pane** (Figure 307).

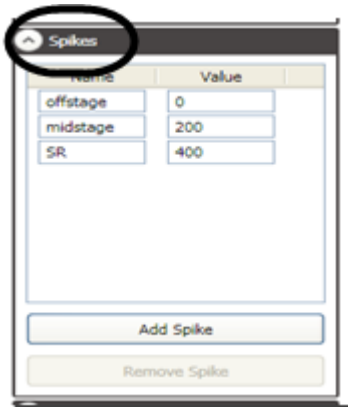


Figure 307

3. Edit the **Value** for the applicable **Spike**, and press the **TAB** key.
*In the example below, the "midstage" Spike **Value** has been changed to "200.25" (Figure 308).*

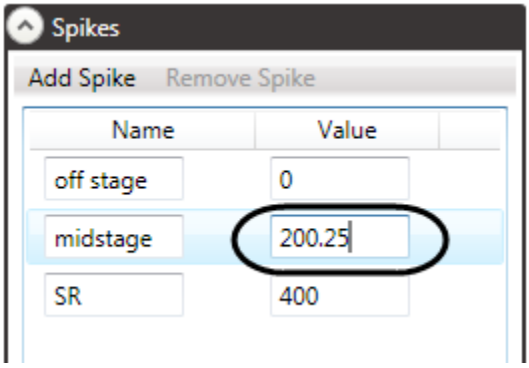


Figure 308

4. The position values will update in the cues that reference the Spike you edited (Figure 309).

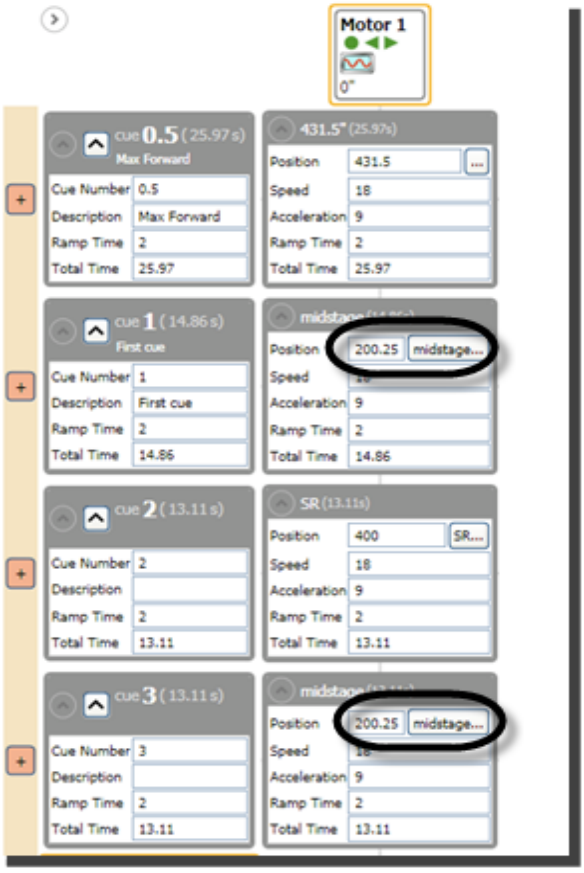


Figure 309

Running Shows

All of the setup, configuring, and cue writing is, in the end, just a prerequisite for what you really need to do—run a performance. As the balance of writing in this manual may indicate, running a show is actually the easiest part of the process. Thankfully, by doing all your preparation attentively, running shows is easy. We'll look at a few ways to execute cues, navigate through the show, and explain what to do when something doesn't work as you expected.

Creating a New Show

When Spikemark first opens, it adds a single motor and a single cue to the show. You can start editing the motor and cue as a starting point for your show. If you would like to begin with a blank slate, you can create a new show by clicking **New** in the **File** menu (Figure 310).

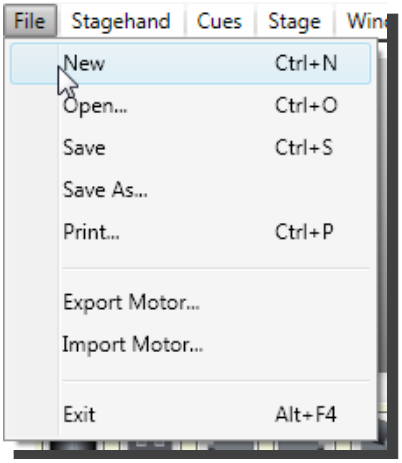


Figure 310

Setting up the Workspace

Once you've finished writing and editing cues, there are a few things you may want to arrange differently in the interface to see the appropriate amount of information. The way in which you like to set up the user interface is a personal preference, so the points outlined here are merely suggestions.

In the steps below, we'll expand the **Cue Grid** to take up the whole screen by double clicking on the splitter bars to minimize the **Stagehand Editing Pane** and the **Stage Model Viewer**. Next we'll add a **Stage Model Viewer** that will sit on top of the Spikemark main screen by clicking **Stage Model Viewer** in the **Window** menu. Resize and arrange to your liking.

1. Expand the **Cue Grid** to its full size by double clicking on the **Stage Model Viewer** and the **Stagehand Editing Pane**'s Splitter Panes (Figure 311).

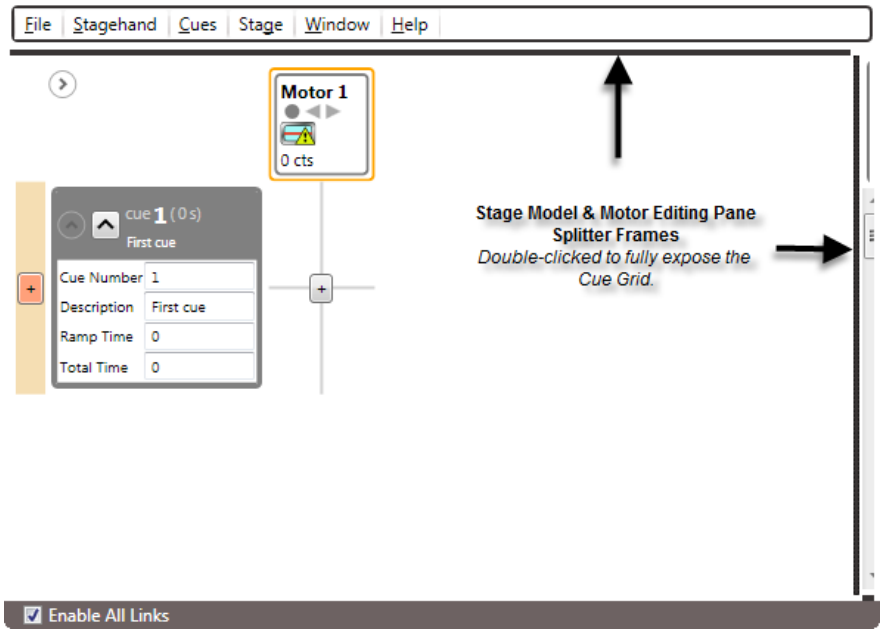


Figure 311

2. Hide all the speed and acceleration information for the cues by right-clicking in the **Cue Grid** and selecting **Collapse All** (Figure 312).
This will compact the Cue Grid while allowing you to see the time each cue will take and the positions of every motor (Figure 313).

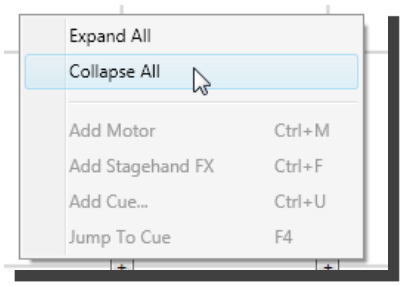


Figure 312

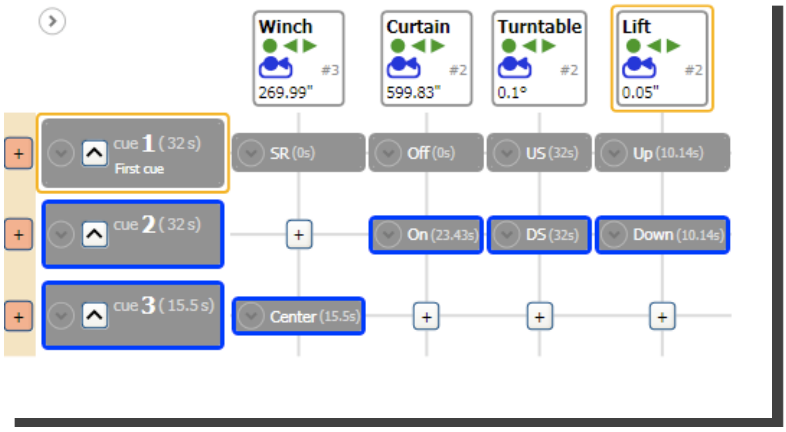


Figure 313

3. In **Window** menu, click **Stage Model Viewer** to create a floating **Stage Model Viewer**. Position the **Stage Model Viewer** in some unoccupied space of the **Cue Grid** (Figure 314) or drag it over to a second monitor.

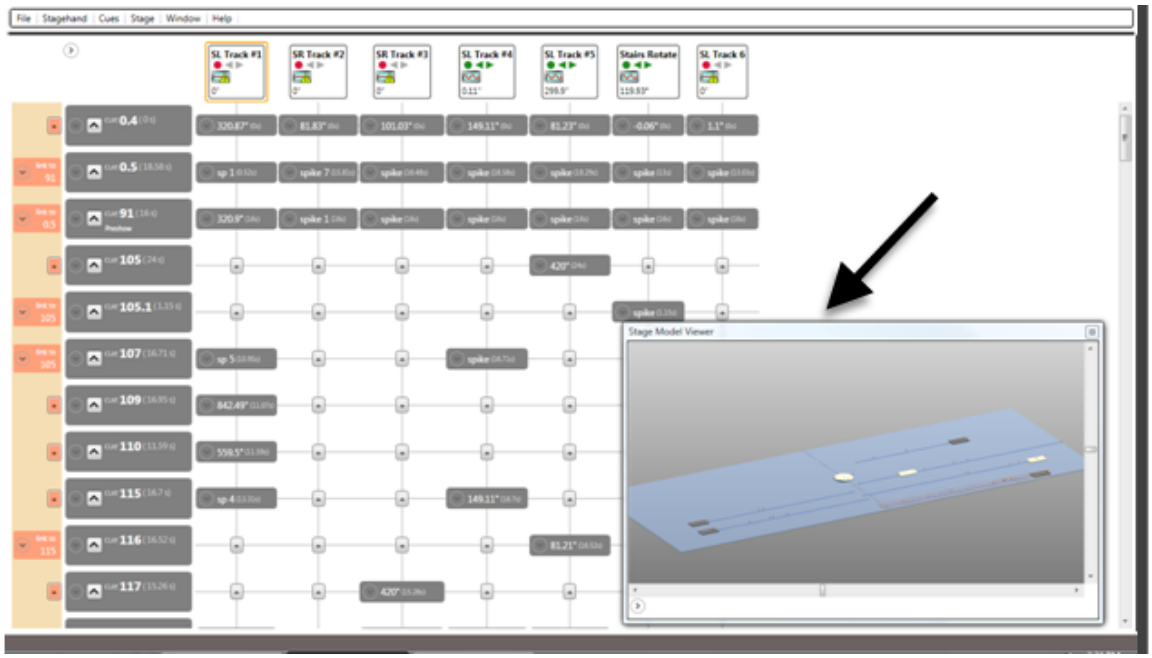


Figure 314

Rearranging the Cue Grid

By default, motors appear as columns in the **Cue Grid** in the order in which they were created. While that's sensible to Spikemark, it probably bears no significance to you. Luckily you can rearrange the columns of the **Cue Grid** to follow the logic of your show.

1. Right-click in a blank section of the **Cue Grid** and select **Rearrange Stagehand Order…** (Figure 315).

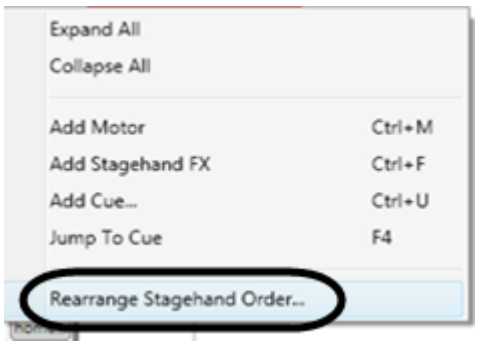


Figure 315

2. In the **Stagehand Display Order Editor** dialog box, select the motor that you wish to move and then use the up/down arrow buttons to rearrange the display order (Figure 316).

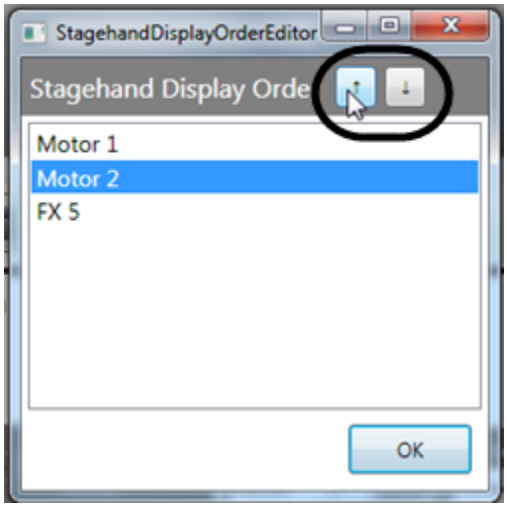


Figure 316

Loading a Cue

Executing a cue is a two step process. The first step is to load the cue. Loading the cue sends all the cue information out to the Stagehand Motor or FX and gets the motors ready to move. Typically, you'll load a cue when the stage manager gives you a "warning" for the cue.

NOTE: Cues that are linked do not need to be explicitly loaded, as the link will *automatically* load and execute its child cue.

There are a few ways to load a cue. The difference is a matter of personal preference.

1. Click the **Load Cue** button in the **Cue Grid** (Figure 317).

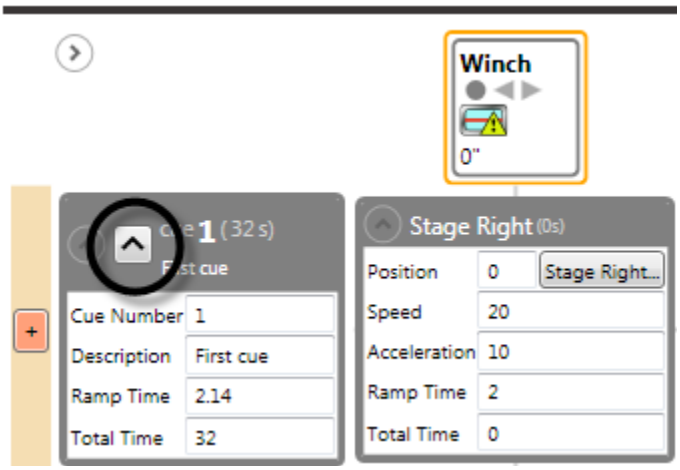


Figure 317

Or, click the **Jump To Cue** button on the **Cue Controller** in the **Stagehand Editing Pane** (Figure 318) and then select a cue from the Jump to Cue dialog box (Figure 319). *You can also type a cue number or description to jump to the cue.*

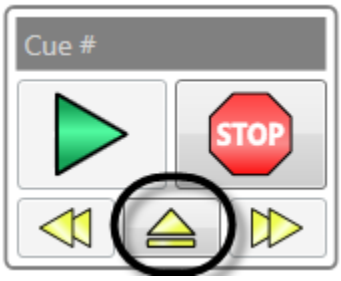


Figure 318

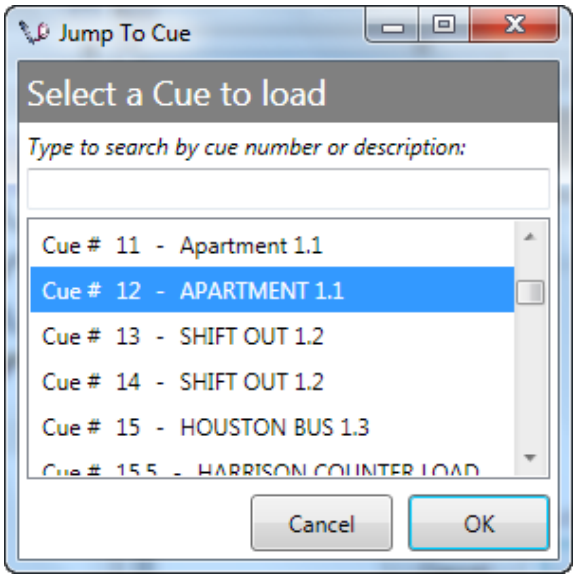

Jump To Cue
Select a Cue to load
Type to search by cue number or description:
Cue # 11 - Apartment 1.1
Cue # 12 - APARTMENT 1.1
Cue # 13 - SHIFT OUT 1.2
Cue # 14 - SHIFT OUT 1.2
Cue # 15 - HOUSTON BUS 1.3
Cancel
OK

Figure 319

Or, if a cue was loaded or run previously, you can use the **Next Cue** and **Previous Cue** buttons in the **Cue Controller** (Figure 320).

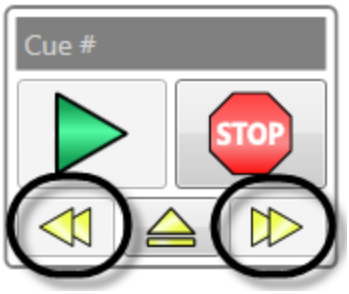


Figure 320

Or, press the **Load Cue** button on your Showstopper and select a cue from **Jump to Cue** dialog box.

Unloading a Cue

After you've loaded a cue, and your hand is hovering of the GO button, you may get the news over your headset that "it's going to be a while". Rather than leave the cue loaded and one accidental click away from running, you may want to unload the cue and return Spikemark to an idle state. There are three ways to unload the currently-loaded cue, use any method that is most convenient:

1. Select **Unload Current Cue** from the **Cues** menu.
2. Right-click on an empty portion of the **Cue Grid** and select **Unload Current Cue** from the pop-up menu.
3. Press the keyboard shortcut **Ctrl + F4.**

Visual Indicators of Primed and Ready Movements

1. In the **Cue Grid**, the borders around the ready cue and ready movements change to red and the Stagehand column headers display a rocket ship countdown icon (Figure 321). *For a complete list of status icons and their meanings, see the "Motor Status" section on page 68 and the "FX Status" section on page 108.*

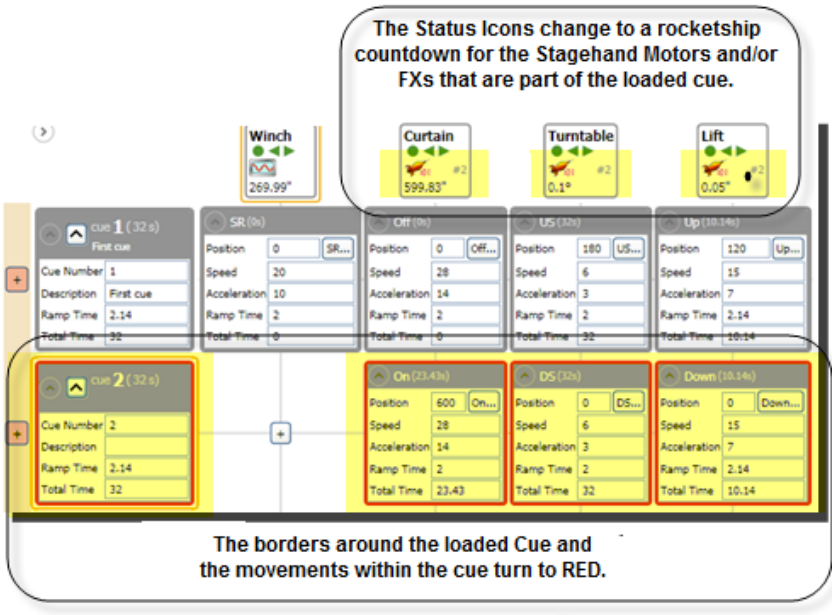

The Status Icons change to a rocketship countdown for the Stagehand Motors and/or FXs that are part of the loaded cue.
Winch
Curtain
Turntable
Lift
The borders around the loaded Cue and the movements within the cue turn to RED.

Figure 321

NOTE: If only some of the Stagehands were able to load the cue, the cue outline will turn pink indicating a partially loaded cue. This can occur if a Stagehand is not currently connected to Spikemark.

2. The status bar text updates in the bottom left corner of Spikemark (Figure 322).

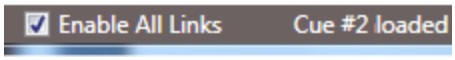


Figure 322

3. The motor(s) that will move in the loaded Cue change to red and their target position(s) are shown in semi-transparent red inside the **Stage Model Viewer** (Figure 323).
Note that the Turntable's target position in the example below is the thin, semi-transparent red line that sits down stage, nearest the lift.

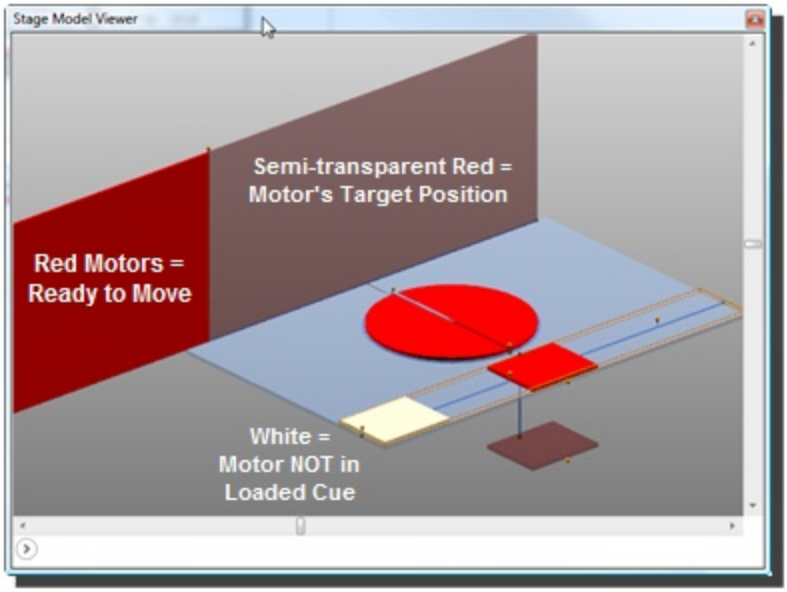


Figure 323

Running a Cue

Once a cue is loaded, you can start all the motors moving by running the cue. There are a few ways to run a cue.

1. You can press the **GO** button in the **Cue Controller** in the **Stagehand Editing Pane** (Figure 324).

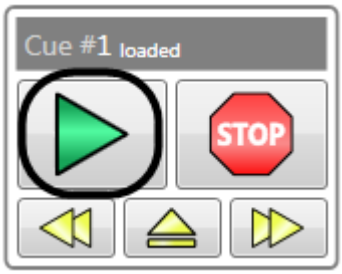


Figure 324

Or, click **Run Cue** in the **Cues** menu (Figure 325).

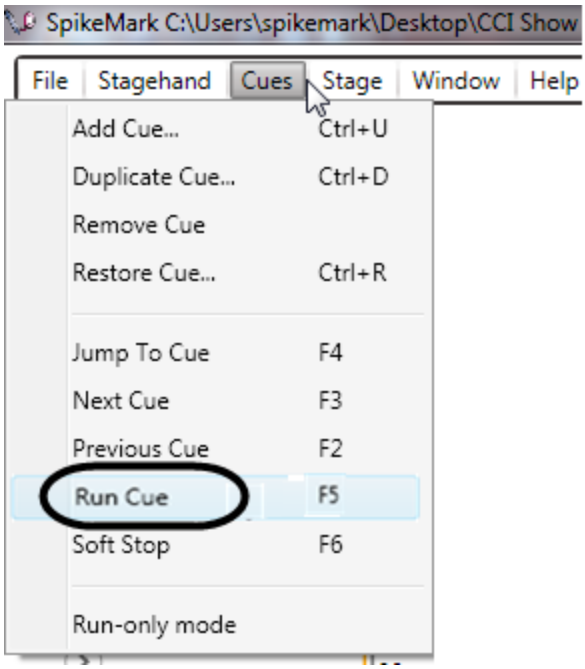


Figure 325

Or, press **F5** on the keyboard.

Or, press the **Go Cue** button on your Showstopper.

What Happens While a Cue Is Running

Once the cue is running, the interface changes to inform you exactly *what* is running.

1. In the **Cue Grid**, the border around the Cue that is running and motor movements within the running Cue change to green and the Stagehand status icons change (Figure 326). *For a complete list of status icons and their meanings, see the "Motor Status" section on page 68 and the "FX Status" section on page 108.*

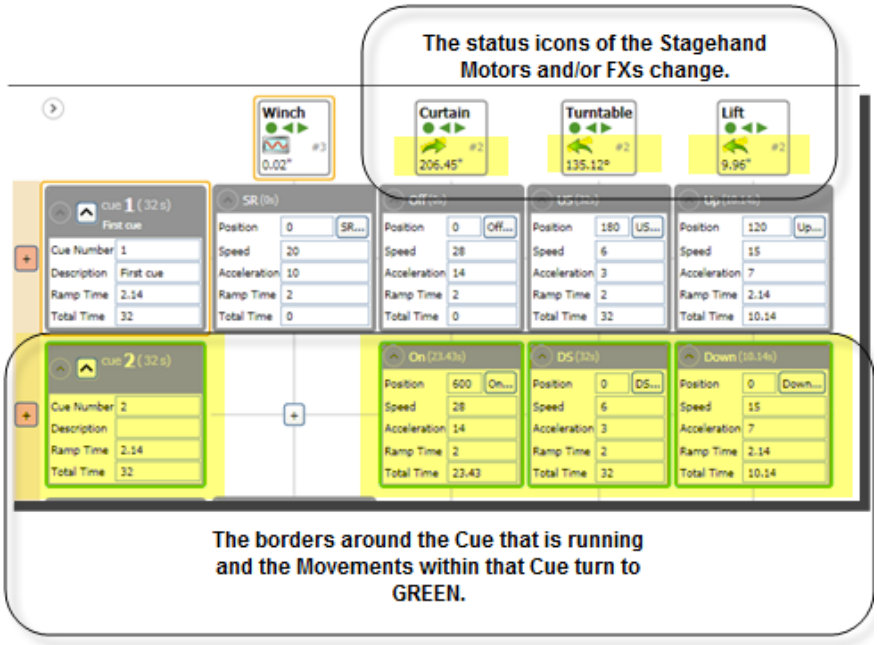

The status icons of the Stagehand Motors and/or FXs change.
Winch
Curtain
Turntable
Lift
The borders around the Cue that is running and the Movements within that Cue turn to GREEN.

Figure 326

2. The motor(s) that are moving in the Cue change to green inside the **Stage Model Viewer** (Figure 327).

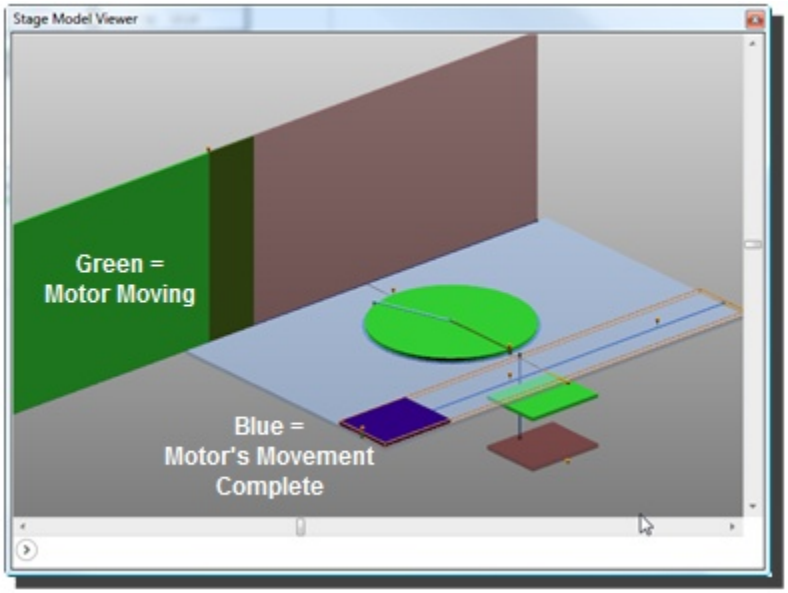


Figure 327

3. The status text in the **Cue Controller** displays which cue is running (Figure 328).

Figure 328

NOTE: If you have *multiple cues* running at the same time, the Cue Controller only shows the last cue started.

4. The **Status Bar** text in the bottom left corner of the Spikemark screen displays which cue is running (Figure 329).

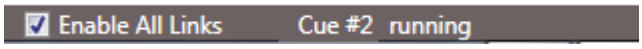


Figure 329

NOTE: If you have *multiple cues* running at the same time, the Status Bar only shows the last cue started.

5. The **Cue** row header in the **Cue Grid** will display a **Countdown Timer** in the upper right corner showing how much time is left until the cue is complete.

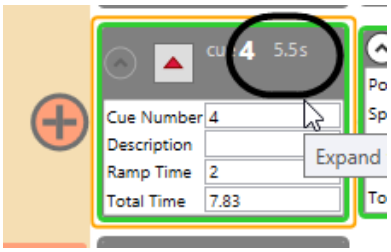


Figure 330

What Happens as Motor Movements and Cue(s) Complete

1. As the motor movements and the cue complete, the interface shifts colors again and the borders in the **Cue Grid** turn blue (Figure 331).

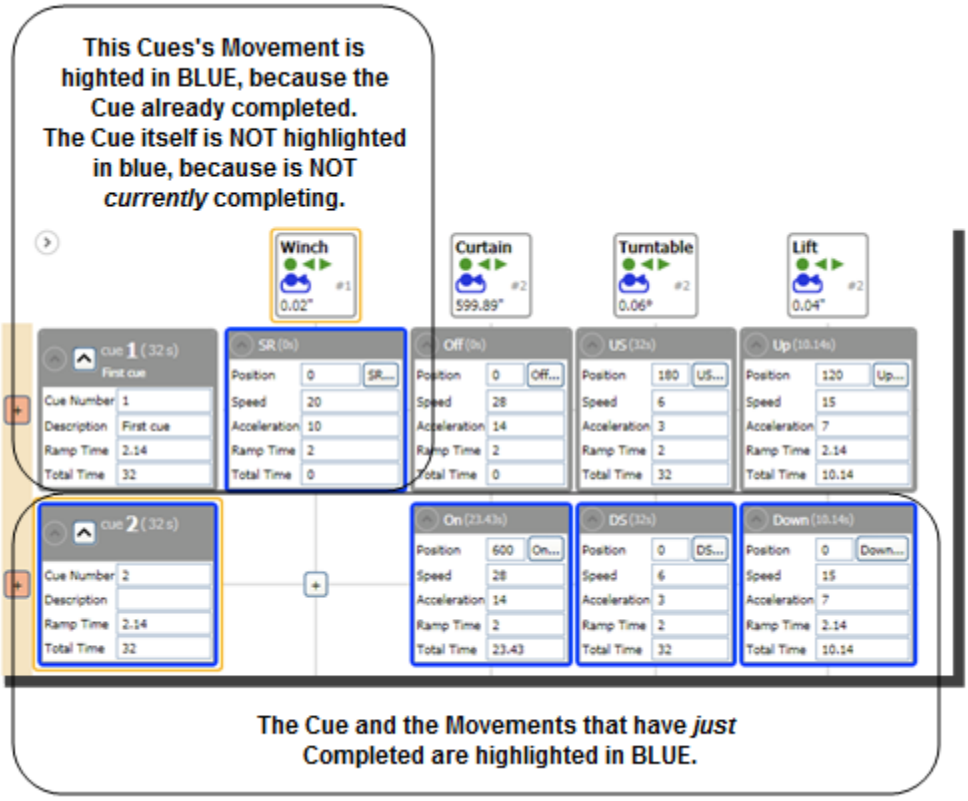

This Cues's Movement is highted in BLUE, because the Cue already completed.
The Cue itself is NOT highlighted in blue, because is NOT *currently* completing.
Winch
Curtain
Turntable
Lift
The Cue and the Movements that have *just* Completed are highlighted in BLUE.

Figure 331

2. The motor(s) that have completed their movements in the Cue change to blue inside the **Stage Model** Viewer (Figure 332).

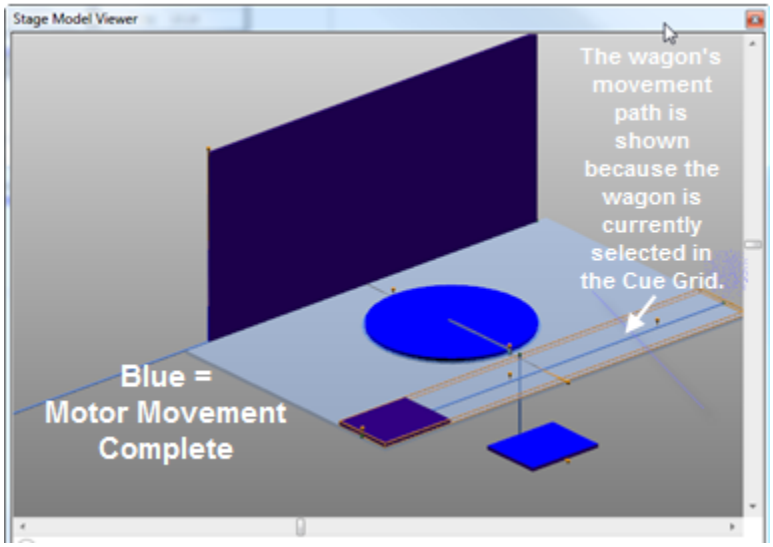

Stage Model Viewer
The wagon's movement path is shown because the wagon is currently selected in the Cue Grid.
Blue = Motor Movement Complete

Figure 332

3. The **Cue Controller** in the **Stagehand Editing Pane** on your screen updates its status text (Figure 333).

Figure 333

4. The **Status Bar** text in the bottom left part of the Spikemark screen is updated (Figure 334).

Figure 334

Deciphering Stagehand Motor Status During a Show

In the heat of a performance, you are likely to feel compelled to obsessively check that your motors and FXs are all connected, happy, and ready to run. There's no shame in that compulsion, it's what makes you a good automation operator. If there is a problem, you want to know about it so that you can fix it quickly and be ready to run your next cue. Spikemark feeds that hunger for status reports with a few key signals in the interface.

The **Cue Grid** header cells give you quick status information about every Stagehand Motor and FX. Let's take a look at a Motor header first (Figure 335).

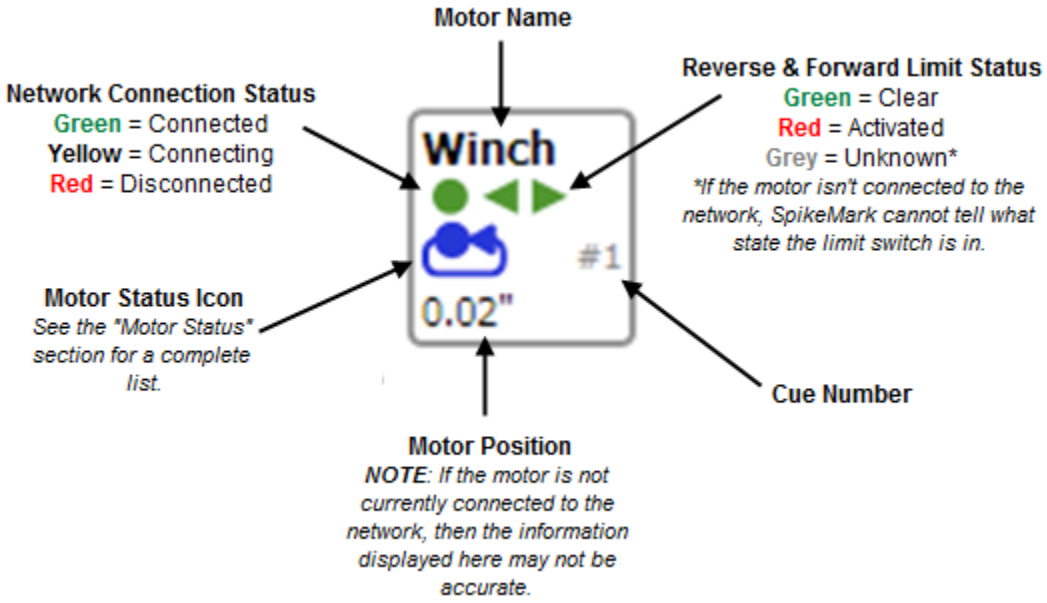

Motor Name
Network Connection Status
Green = Connected
Yellow = Connecting
Red = Disconnected
Reverse & Forward Limit Status
Green = Clear
Red = Activated
Grey = Unknown*
*If the motor isn't connected to the network, SpikeMark cannot tell what state the limit switch is in.
Winch
#1
0.02"
Motor Status Icon
See the "Motor Status" section for a complete list.
Cue Number
Motor Position
NOTE: If the motor is not currently connected to the network, then the information displayed here may not be accurate.

Figure 335

If the motor has been Deactivated (by deselecting the **Active** box in the **Cue Behavior** section of the **Stagehand Editing Pane**), the background of the motor and all its movements will turn grey (Figure 336).

Figure 336

The **Cue Grid** header cells give you quick status information about every Stagehand FX (Figure 337).

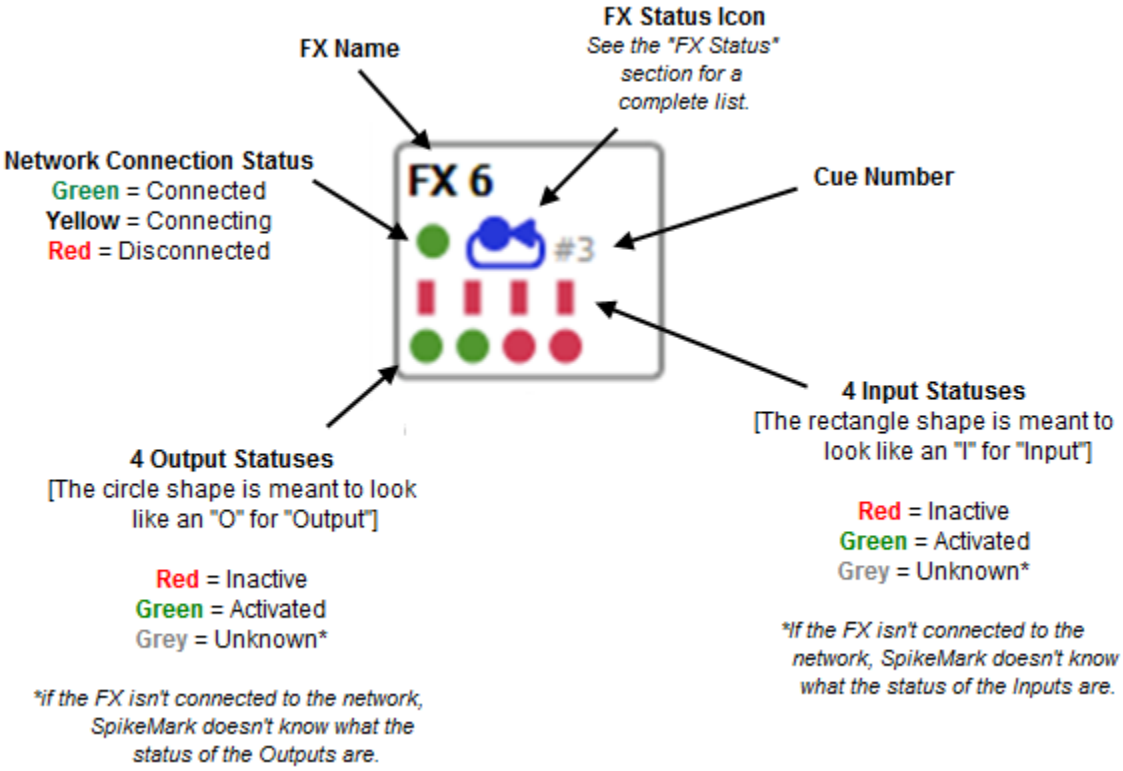

FX Name
FX Status Icon
See the "FX Status" section for a complete list.
Network Connection Status
Green = Connected
Yellow = Connecting
Red = Disconnected
FX 6
#3
Cue Number
4 Input Statuses
[The rectangle shape is meant to look like an "I" for "Input"]
Red = Inactive
Green = Activated
Grey = Unknown*
*If the FX isn't connected to the network, SpikeMark doesn't know what the status of the Inputs are.
4 Output Statuses
[The circle shape is meant to look like an "O" for "Output"]
Red = Inactive
Green = Activated
Grey = Unknown*
*if the FX isn't connected to the network, SpikeMark doesn't know what the status of the Outputs are.

Figure 337

If the Stagehand FX has been Deactivated (by deselecting the **Active** box in the **Cue Behavior** section of the **FX Editing Pane**), the background of the motor and all its movements will turn grey (Figure 338).

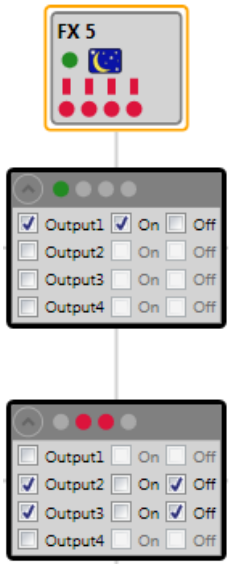


Figure 338

Restoring a Cue

Spikemark's **Cue Restore** feature allows you to move backwards through a show during rehearsal. **Cue Restore** will put every motor back into its proper position for a specific cue, regardless of whether the motor was moving in that cue or had its last movement in a previous cue. If that sounds confusing, it is straightforward in practice. Let's take a look at how the feature works and you'll see that it does exactly what you need when jumping backwards in rehearsal.

In the following example, assume that we just ran through Cue #4 and the stage manager needs to re-run Cue #4 because Jeff, the lighting designer, needs to look at the transition again. As you glance at the cue sequence below, you can see that we need to reset the stage so that we are sitting in Cue #3. But there's a problem. You can't just run Cue #3 to reset. If you were to just run Cue #3, then Motor 1 would be left in its "home" position. So you would need to run Cue #3 and Cue #2. Not a big deal in this scenario, but as you have more motors and more complex cues, restoring gets to be a real pain. Thankfully, **Cue Restore** can handle this easily.

1. Right-click on Cue #3 and select **Restore** (Figure 339).

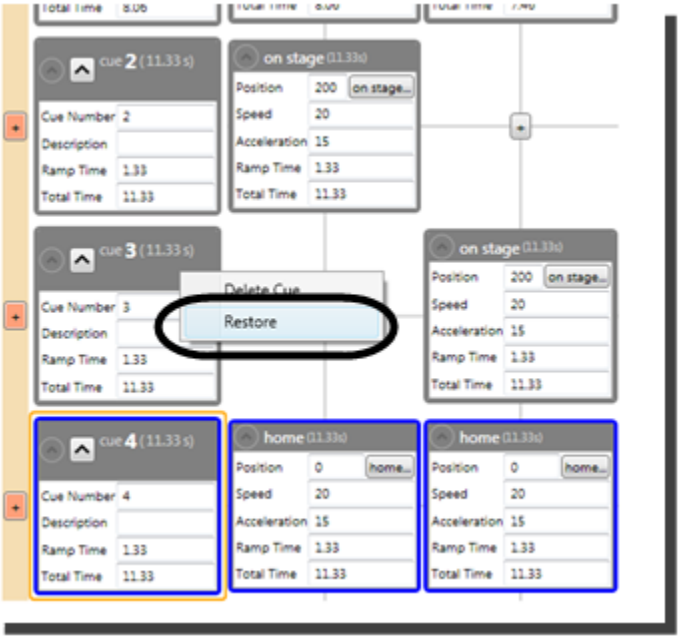


Figure 339

2. A list is presented showing that both **Motor 1** and **Motor 2** need to move in order to restore Cue #3 (Figure 340).

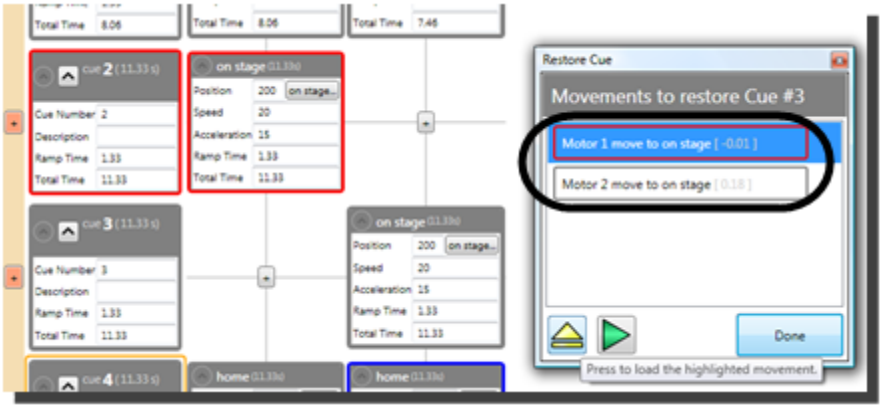

Restore Cue
Movements to restore Cue #3
Motor 1 move to on stage [-0.01]
Motor 2 move to on stage [0.18]
Done
Press to load the highlighted movement.

Figure 340

NOTE: The checkbox in the status bar at the bottom of the Spikemark main screen—"Enable All Links"—can be used to turn off cue links. The Cue Restore feature *automatically* deselects the Enable All Links field [pictured below] in the status bar at the bottom of the Spikemark main screen, so that cue links are not accidentally triggered during a Cue Restore.

3. To start moving motors back into their correct positions, select a single movement from the list and then press the **Load Movement** button in the **Restore Cue** window (Figure 341). Load and run the movements one by one.

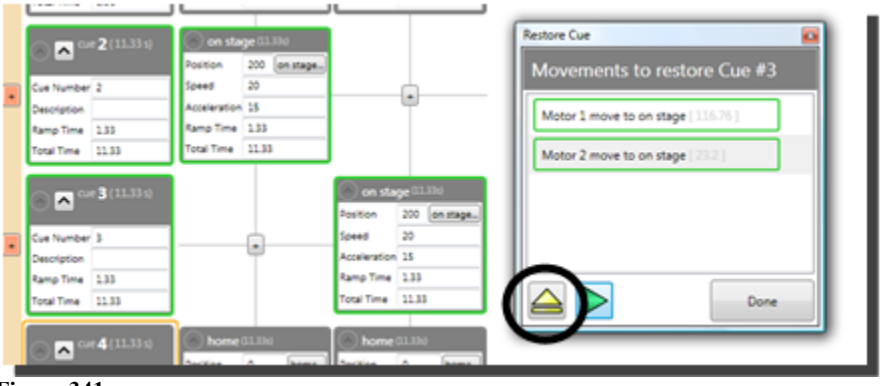


Figure 341

NOTE: You don't have to wait for each movement to complete before starting the next movement. But you must confirm that restoring a movement *will not cause a collision on stage.*

4. After one movement is loaded, you can press the **Run** button in the **Restore Cue** window (Figure 342). Then—*if it is safe*—load the next movement and run it as well.
   In the example below, two movements are running concurrently by starting Motor 2 before Motor 1 completes.

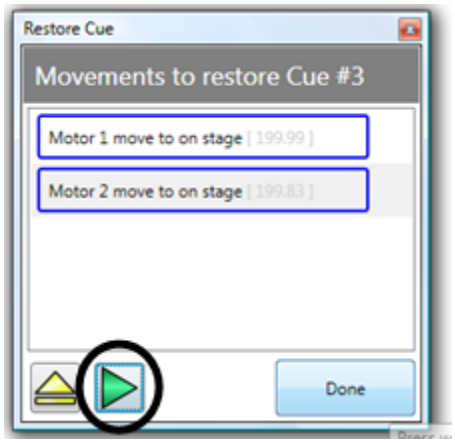


Figure 342

5. When you are finished restoring all the motors, press the **Done** button to close the **Restore Cue** window (Figure 343).

Figure 343

6. Click **OK** in the **Enable Cue Links** dialog box to re-enable all cue links (Figure 344).
   *If the motors have NOT finished moving, click **Cancel**.*

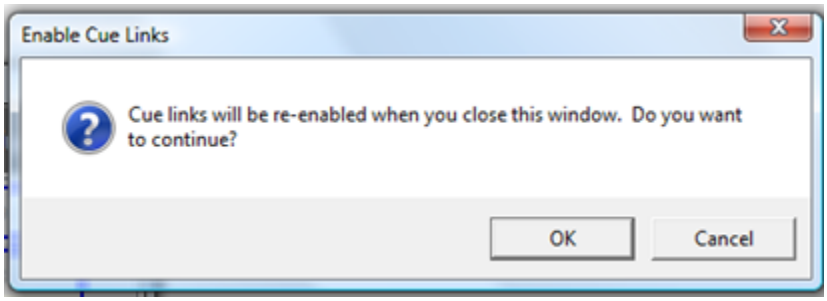


Figure 344

7. Notice that in the status bar in the bottom left-hand corner of the Spikemark main screen, the **Enable All Links** box is once again selected (Figure 345).

Figure 345

8. Cue #4 is restored and ready to run again. Repeat the steps to restore the cue again, if needed.

Dataton WATCHOUT Integration – *New in Spikemark 3!*

Projection design is commonplace in an increasing number of theatrical performances. With media servers and automation gear existing on the same network, it seems obvious that these systems should share data to create stunning effects by coordinating video and scenic motion. To that end, Spikemark now has the ability to output position data from any motor in a show to a WATCHOUT media server. It takes a little extra configuration in Spikemark and WATCHOUT to get both systems talking to each other, but the effort is reward with truly stunning effects. Alright, let's get started.

Assume that we have a little show with an automated wall panel attached to a traveler track. On cue, the wall panel will track from Stage Right to Stage Left. As the panel tracks across the stage, we need to project a graphic onto the panel and have the image move along as if it were glued to the panel. Here's a screen shot of the Spikemark cue (Figure 346):

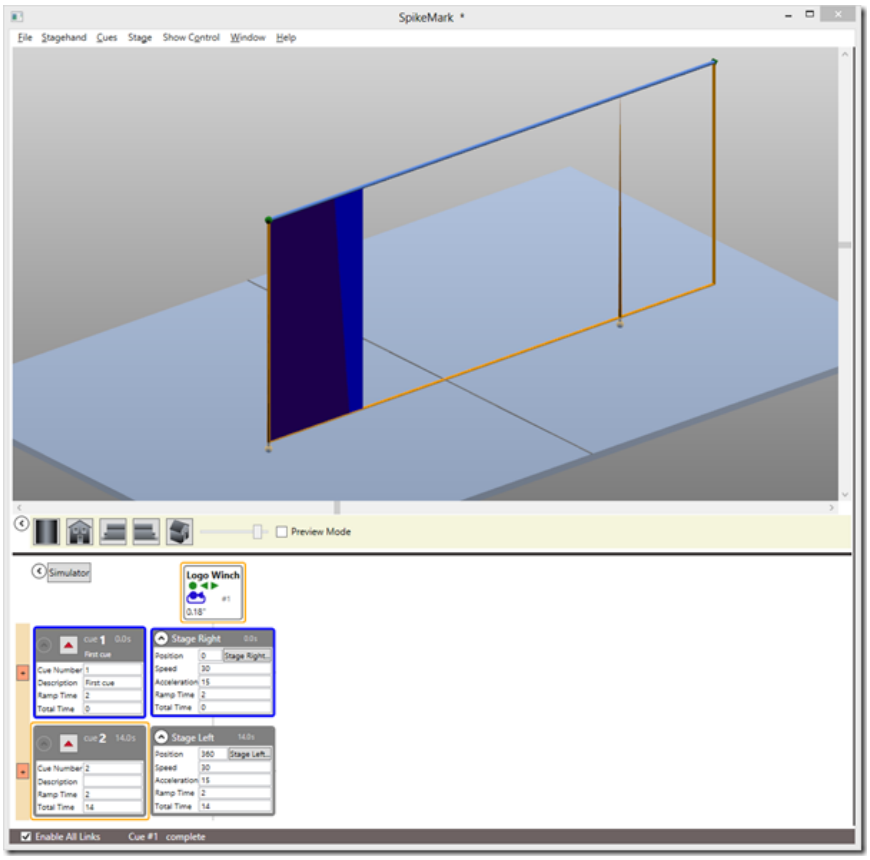


Figure 346

In order for WATCHOUT's projectors to track an image synchronously with the motorized panel, we need to send the position of the panel to WATCHOUT. Spikemark will communicate with WATCHOUT over the Ethernet network, so both the Spikemark automation computer and the WATCHOUT production computer need to be plugged into the same physical network. In

addition, the two computers need to have compatible IP Addresses that share the first three segments of the address with unique fourth segments. I have the addresses assigned as such: Spikemark computer is 192.168.10.119 (Figure 347)

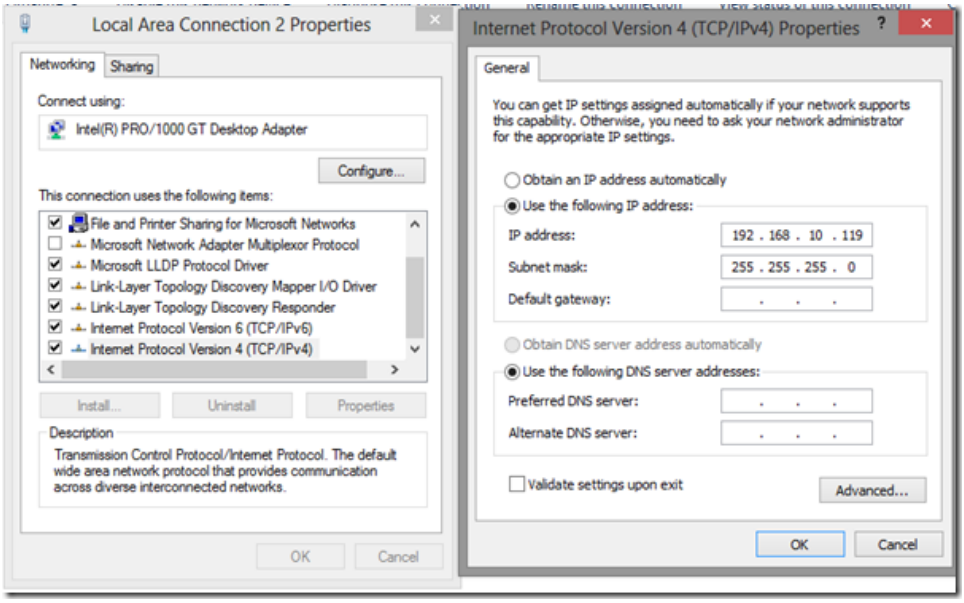

Local Area Connection 2 Properties
Networking
Sharing
Connect using:
Intel(R) PRO/1000 GT Desktop Adapter
Configure...
This connection uses the following items:
File and Printer Sharing for Microsoft Networks
Microsoft Network Adapter Multiplexor Protocol
Microsoft LLDP Protocol Driver
Link-Layer Topology Discovery Mapper I/O Driver
Link-Layer Topology Discovery Responder
Internet Protocol Version 6 (TCP/IPv6)
Internet Protocol Version 4 (TCP/IPv4)
Install...
Uninstall
Properties
Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.
OK
Cancel
Internet Protocol Version 4 (TCP/IPv4) Properties
General
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Obtain an IP address automatically
Use the following IP address:
IP address:
192 . 168 . 10 . 119
Subnet mask:
255 . 255 . 255 . 0
Default gateway:
Obtain DNS server address automatically
Use the following DNS server addresses:
Preferred DNS server:
Alternate DNS server:
Validate settings upon exit
Advanced...
OK
Cancel

Figure 347

And the WATCHOUT computer is 192.168.10.9 (Figure 348)

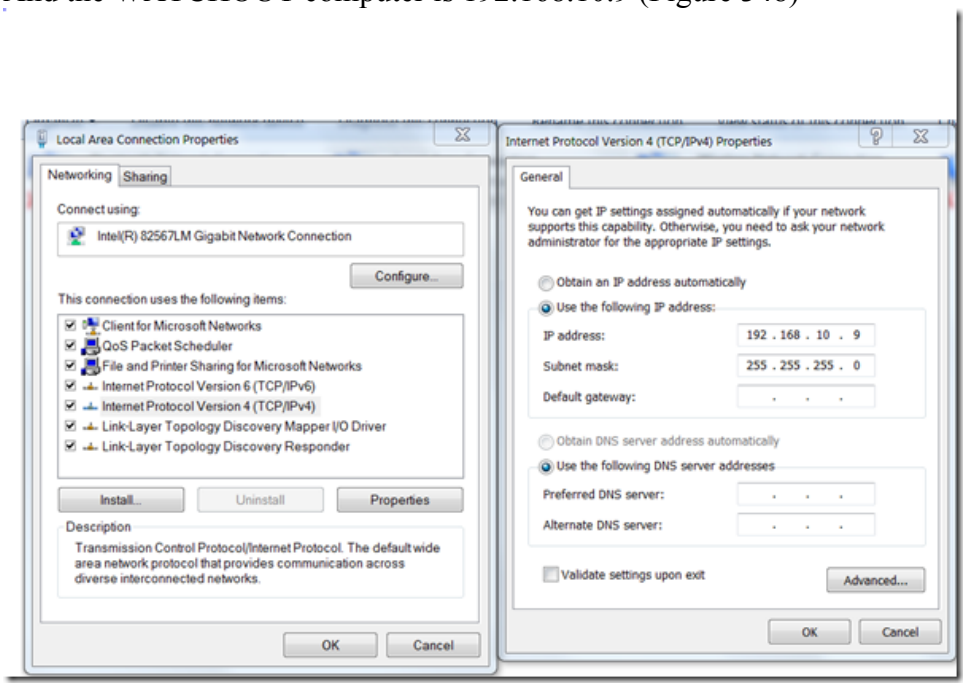

Local Area Connection Properties
Networking
Sharing
Connect using:
Intel(R) 82567LM Gigabit Network Connection
Configure...
This connection uses the following items:
Client for Microsoft Networks
QoS Packet Scheduler
File and Printer Sharing for Microsoft Networks
Internet Protocol Version 6 (TCP/IPv6)
Internet Protocol Version 4 (TCP/IPv4)
Link-Layer Topology Discovery Mapper I/O Driver
Link-Layer Topology Discovery Responder
Install...
Uninstall
Properties
Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.
OK
Cancel
Internet Protocol Version 4 (TCP/IPv4) Properties
General
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Obtain an IP address automatically
Use the following IP address:
IP address:
192 . 168 . 10 . 9
Subnet mask:
255 . 255 . 255 . 0
Default gateway:
Obtain DNS server address automatically
Use the following DNS server addresses
Preferred DNS server:
Alternate DNS server:
Validate settings upon exit
Advanced...
OK
Cancel

Figure 348

With both computers addressed properly, we need to tell Spikemark where to send the position data. In Spikemark select **Watchout** from the **Show Control** menu (Figure 349).

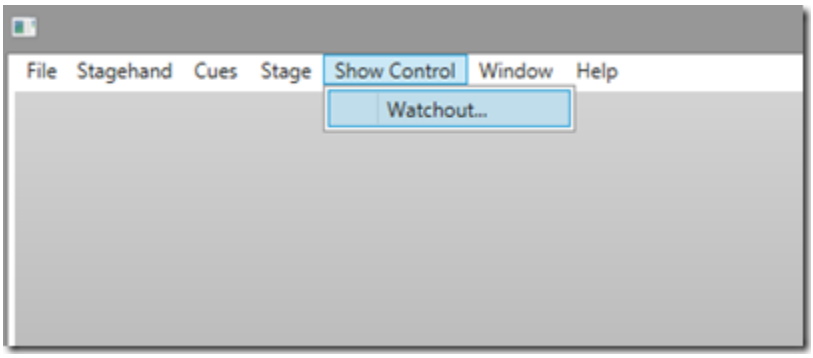


Figure 349

A dialog pops up with some configuration details that determine what data is sent to WATCHOUT (Figure 350).

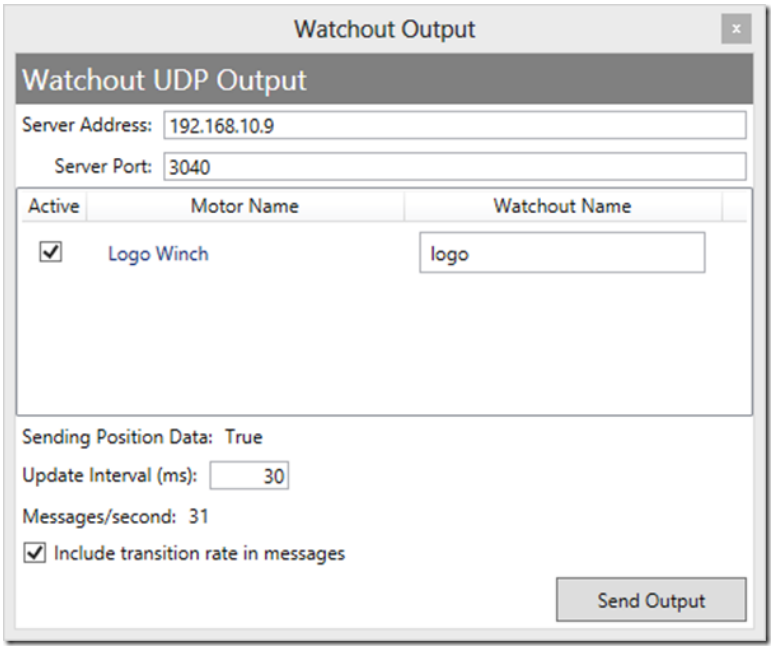


Figure 350

From the top the options are:

1. **Server Address**: The IP Address of the WATCHOUT production computer
2. **Server Port**: The port where WATCHOUT listens for incoming data. By default, WATCHOUT listens on 3040.

3. **Motor List**: Each motor in your show is listed. If the **Active** box is checked, that motor's position information will be sent to WATCHOUT. In the **Watchout Name** text box you can enter a name that will be used inside WATCHOUT to identify the motor's position. The Spikemark motor name and the WATCHOUT name can map however you like. For example, we could have called it "Logo Winch" in Spikemark and "fuzzy pink rabbit" in Watchout.

4. **Sending Position Data**: Indicates whether Spikemark is currently sending UDP packets to the address indicated in **Server Address**.

5. **Update Interval (ms)**: The frequency with which Spikemark will send position updates to WATCHOUT. The number entered here will determine how many milliseconds should elapse between updates, so higher numbers will result in a slower update cycle. In practice, 30ms is about the fastest rate consistently possible without adversely affecting Spikemark's performance.

6. **Messages/second**: The number of position updates that are actually being sent to WATCHOUT each second. This number will often bounce around by 1 or 2 messages.

7. **Include transition rate in messages**: If checked, Spikemark will send WATCHOUT the number of milliseconds that have elapsed since the last position update. WATCHOUT can use this information to smooth the animation of the image as it tracks with the motor. This generally results in a smoother visual result, but can be slightly inaccurate. Feel free to experiment with either setting to get the most appropriate result for your show.

8. **Send Output**: This is a toggle button to turn on/off the data stream from Spikemark. The data is sent via UDP, which is a connectionless protocol, so there is no harm sending out the packets even if WATCHOUT is disconnected from the network. UDP packets will blissfully fall into oblivion if the server is not around to receive them so you can start the output stream before WATCHOUT is running.

With Spikemark configured and the **Send Output** button depressed, we are ready to fire up the WATCHOUT production machine. Start WATCHOUT and give your show file a name. As I mentioned when we started, I need an image to be projected on the traveler panel, so our first step in WATCHOUT is to import an image (Figure 351).

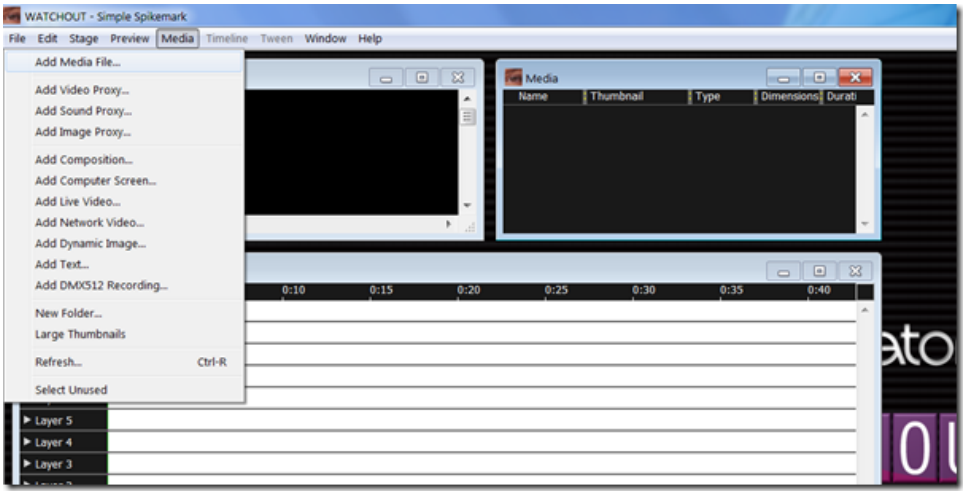


Figure 351

I selected a Creative Conners Logo image, which shows up in the **Media** list (Figure 352).

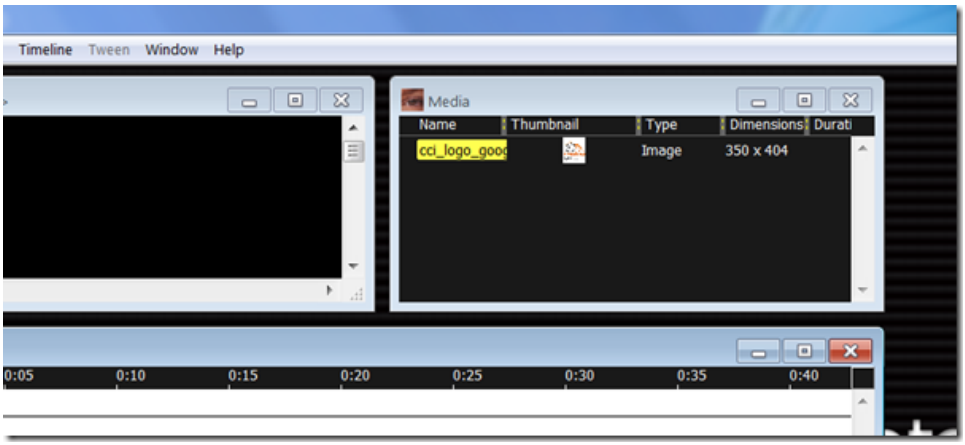


Figure 352

Now drag the image from the **Media** list into the **Stage** window. You can see the image displayed in the center of the **Stage** view and it also shows up in the **Main Timeline** (Figure 353).

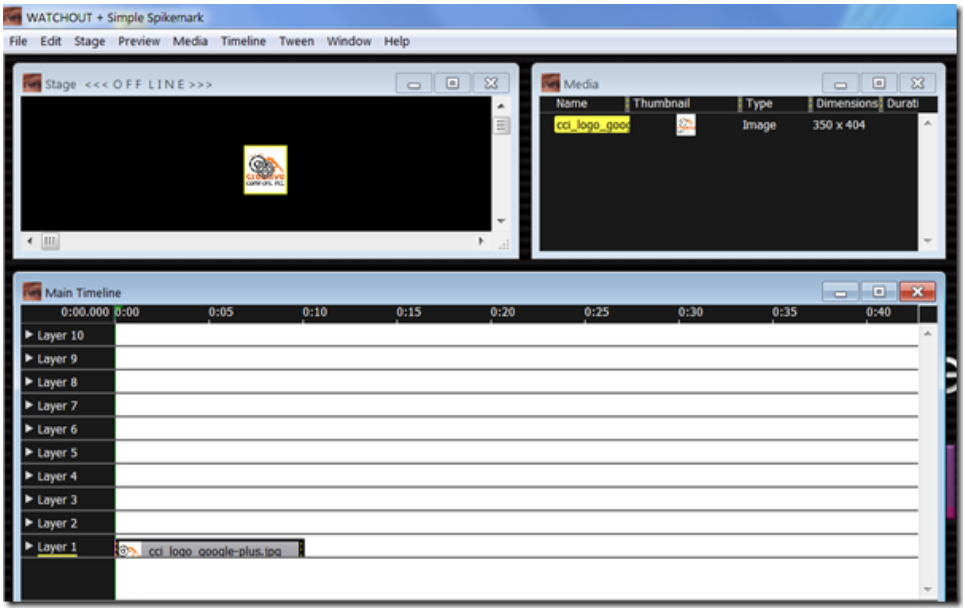


Figure 353

With the image on our virtual stage, we need start configuring WATCHOUT to listen for data from Spikemark. We have to enable an external source (Spikemark in this case) to control the image position. Double-click on the image in the **Media List** and select **More Effects and Capabilities** (Figure 354).

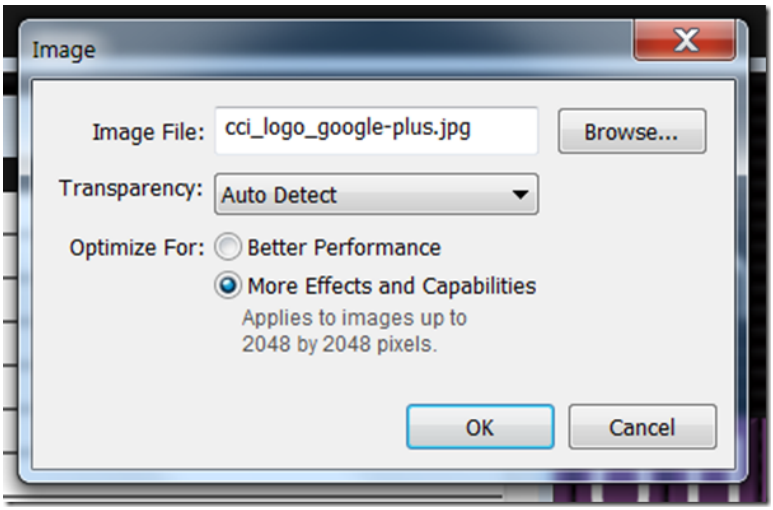


Figure 354

Then, in the **Main Timeline**, double-click on the image layer to bring up the **Media Cue** properties window. Select the **Advanced** tab and check **External Control of Position, Scale & Rotation** (Figure 355)**.**

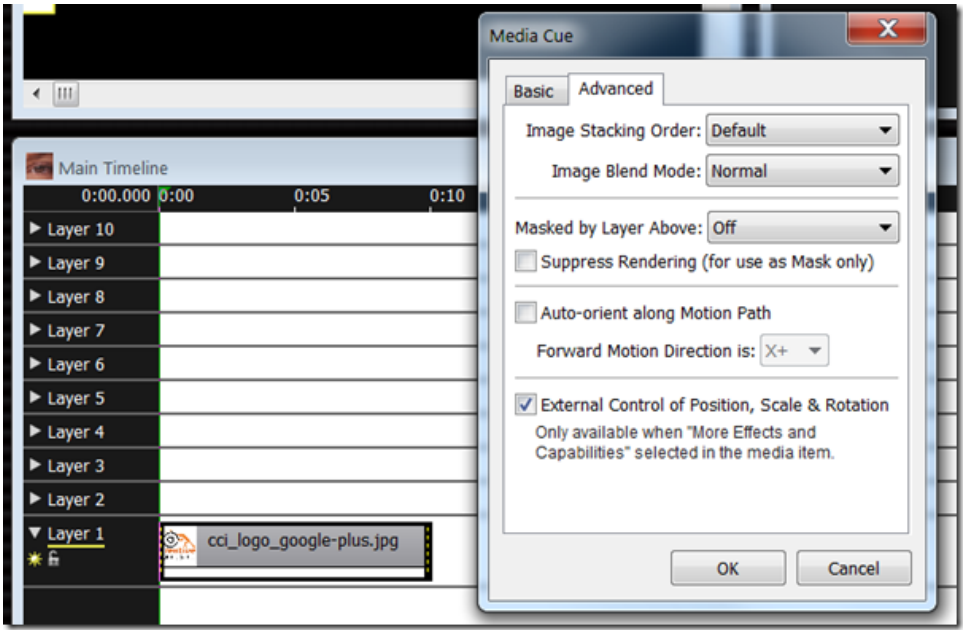


Figure 355

Our next step is to create a **Generic Input** in WATCHOUT that has a name that matches the **Watchout Name** we entered into Spikemark. We will use the data received from that **Generic Input** to move the image around. To add a **Generic Input** select **Input** from the **Window** menu (Figure 356).

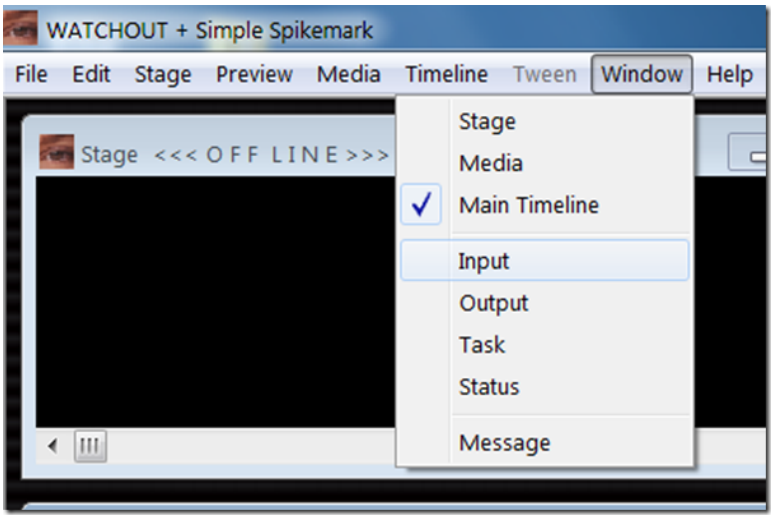


Figure 356

From the **Input** window, click on the little triangle in the upper right corner. From the menu that appears, select **Add Generic Input** (Figure 357).

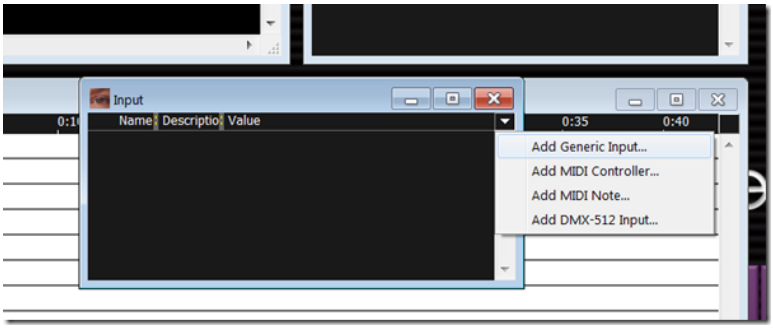


Figure 357

A dialog is presented where you can enter the **Name** of the input and the **Limit** of the input value. This step is important to get correct. The **Name** needs to match the name entered in the Spikemark **Watchout Output** window… exactly, same case, same spelling, etc. The **Limit** should match the highest value expected to come from Spikemark. In this case, our traveler has a maximum forward position of 360", so we can enter 360 (Figure 358).

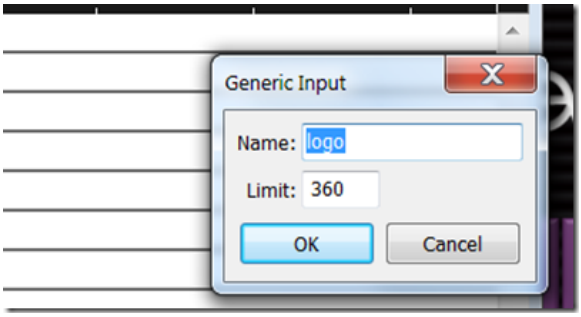


Figure 358

Press the OK button and then the new Generic Input will be listed in the Input list with its current value set to 0.00 (Figure 359).

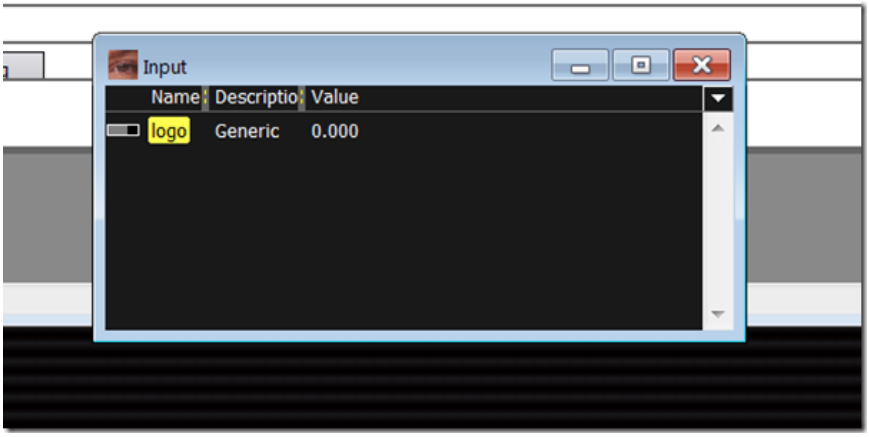


Figure 359

Now, the next step is to connect the value of the **Generic Input** to the x-axis of the Image so that the Image will move as the **Generic Input** value changes. To link the image position to the **Generic Input** value we will create a formula in the **Main Timeline**. Select the image layer in the **Main Timeline**, and then from the **Tween** menu select **Position** (Figure 360).

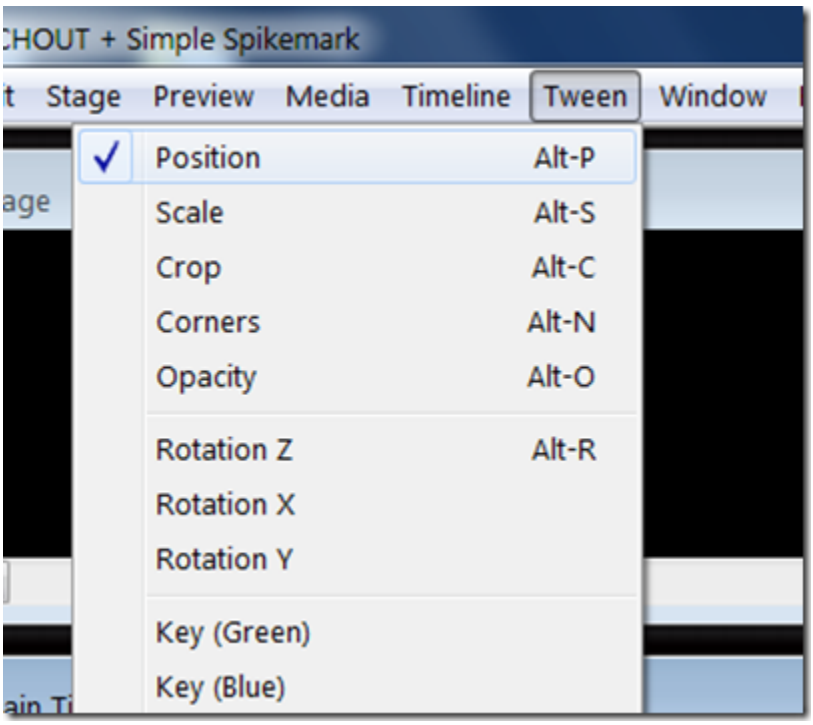


Figure 360

This adds a **Position** tween track below Layer 1 in the **Main Timeline**. This is the good part. Now that we have a tween for Position, on the left side of the track there is a little round button with an "f" inside. That allows us to write a formula that will link the position of the image to the value of the **Generic Input**, the value of the **Generic Input** will be connected to the data stream

from Spikemark, and the data stream from Spikemark is driven by the position of the scenery. The knee bone is connected to the leg bone… still with me? Great, click the little *f*unction button (Figure 361).

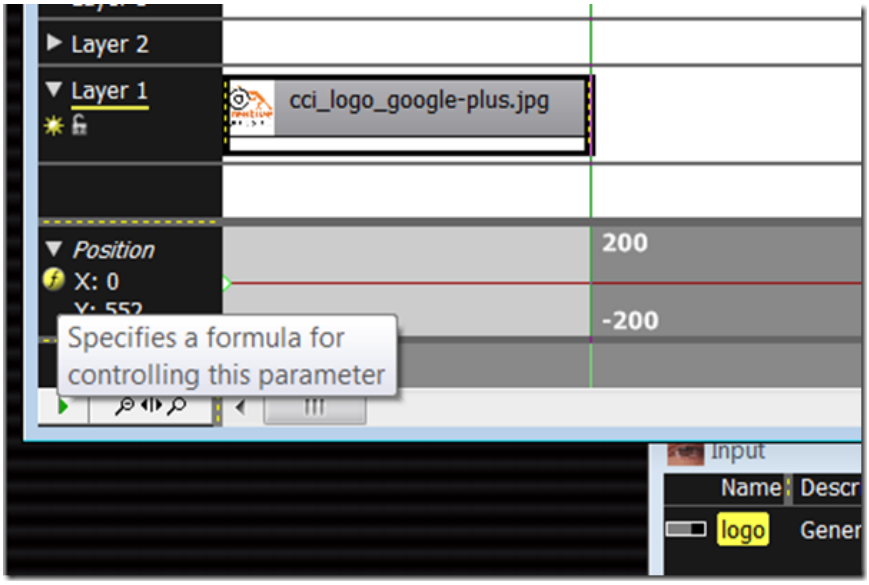


Figure 361

In the dialog box that appears, we enter in a formula in the X axis text box. Since this is a traveler track, we want to manipulate the lateral position of the image, but if it was a flying piece of scenery we could instead control the Y axis of the image. To use the value of the **Generic Input**, we simply type the name of the input. In this case, I'm multiplying the value of the input by 10 to get the image to track the correct number of pixels across the stage. The multiplier you use can be adjusted to fit the specific show (Figure 362).

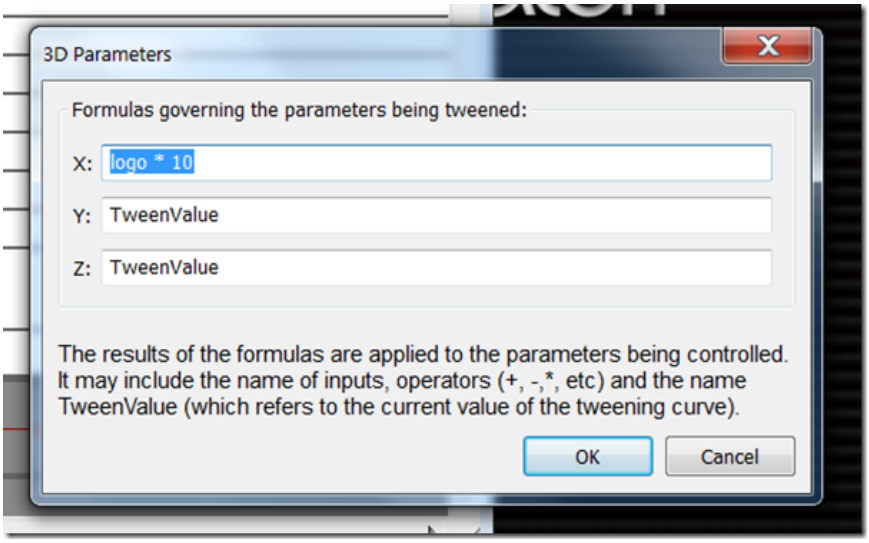

3D Parameters
Formulas governing the parameters being tweened:
X: logo * 10
Y: TweenValue
Z: TweenValue
The results of the formulas are applied to the parameters being controlled.
It may include the name of inputs, operators (+, -,*, etc) and the name
TweenValue (which refers to the current value of the tweening curve).
OK
Cancel

Figure 362

We are almost there. Before flipping the last switch to connect Spikemark to WATCHOUT, try clicking around in the **Value** column of the **Input** list. This will manually adjust the value of the

Generic Input and if everything is correct so far, as you alter the **Generic Input Value** the image should jump to a new X position in the **Stage** window (Figure 363).

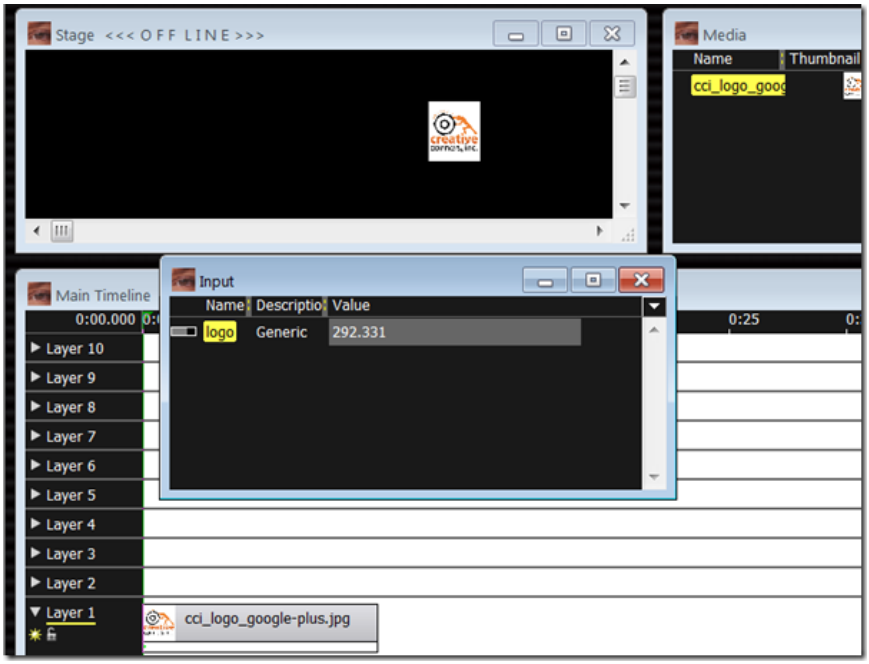


Figure 363

Alright, let's get the WATCHOUT computer listening to the Spikemark computer. From the **File** menu select **Preferences**. Then from the **Control** tab, check the **UDP** box next to **Production Computer Control** (Figure 364).

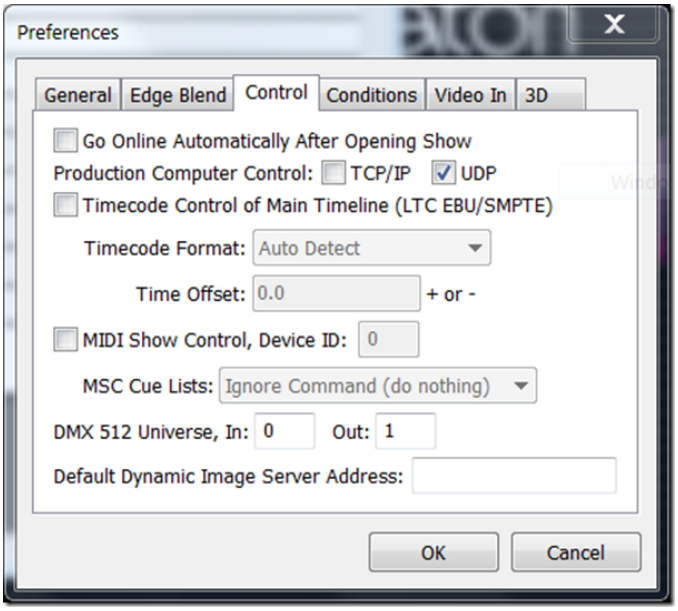


Figure 364

As soon as you click **OK**, WATCHOUT will start picking up the position data stream from Spikemark (assuming you depressed the **Send Output** button in Spikemark) and the image will

snap back to match its X position with the motor position. Also, the **Generic Input Value** should track with the motor position.
Let's load up cue #2 in Spikemark. Notice that the current motor position is 0.18" in Spikemark, and that the Generic Input Value in WATCHOUT is 0.175 showing that the two systems are communicating (Figure 365, Figure 366).

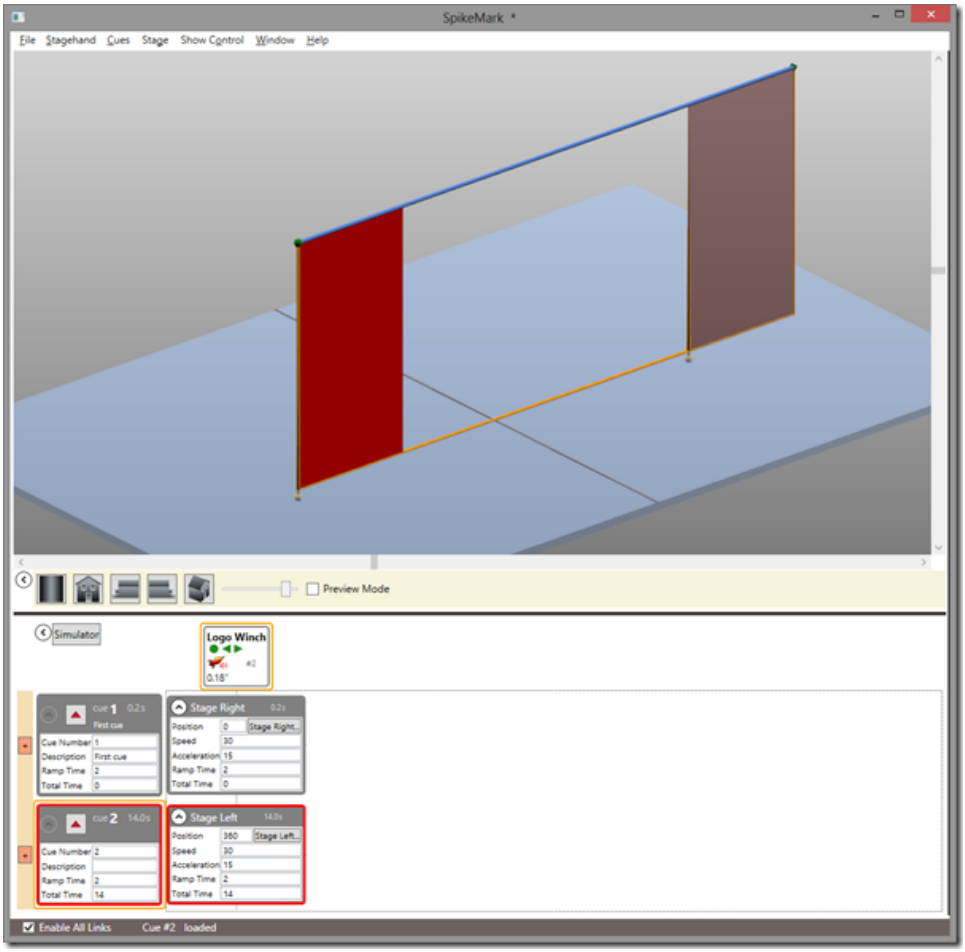


Figure 365

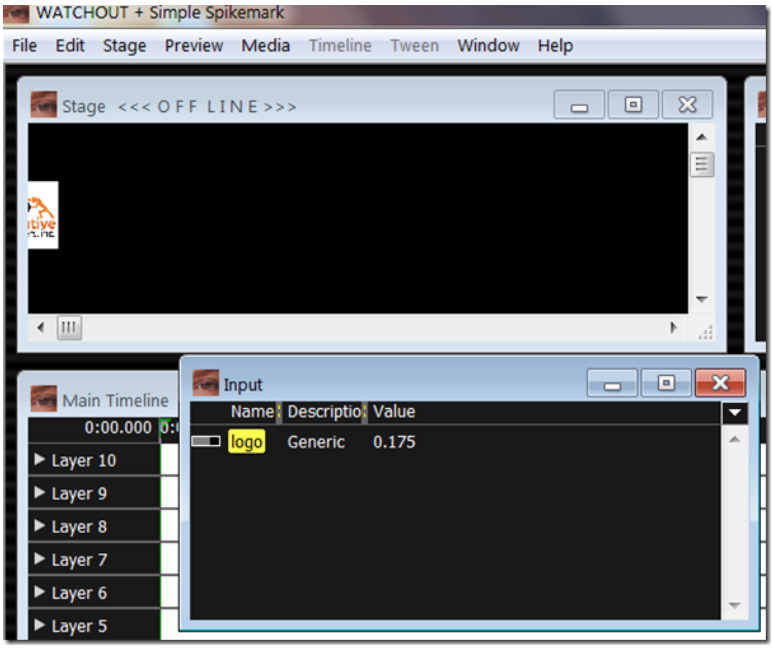


Figure 366

Now, let's run cue #2 in Spikemark (Figure 367). When it completes, we can see that the image tracked across the stage in WATCHOUT, matching the motor position! (Figure 368)

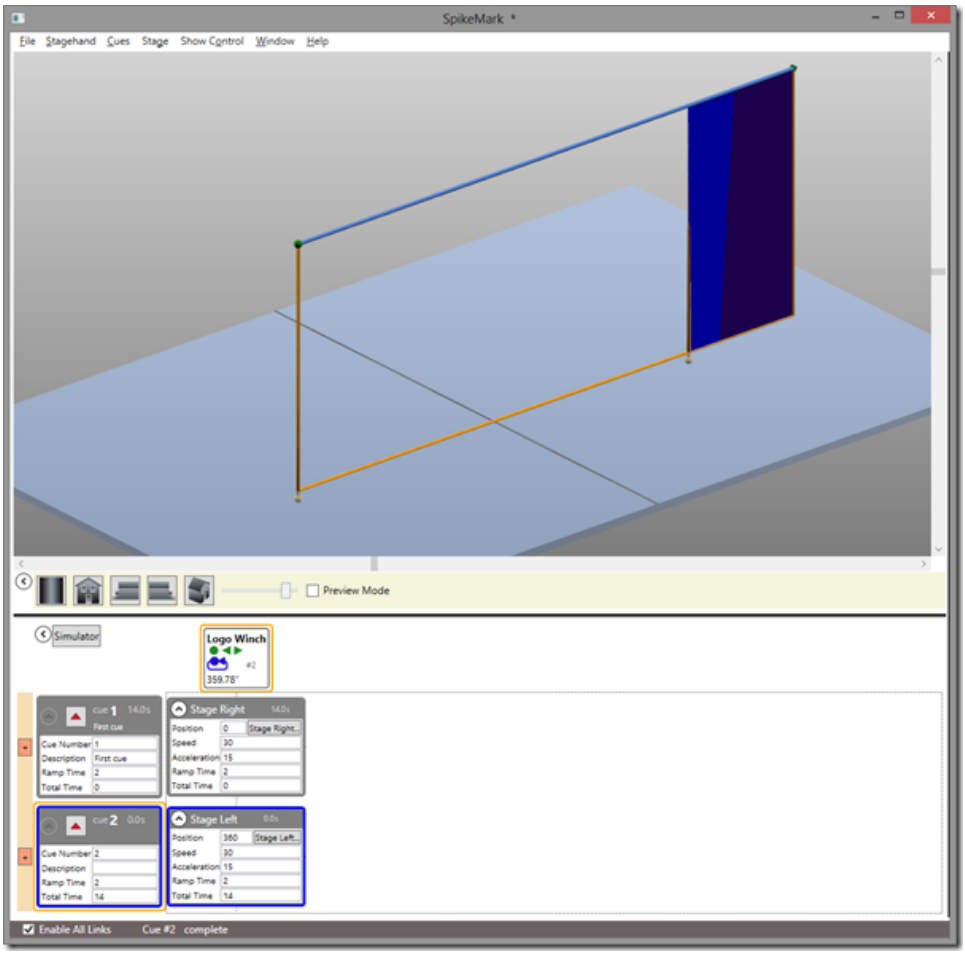


Figure 367

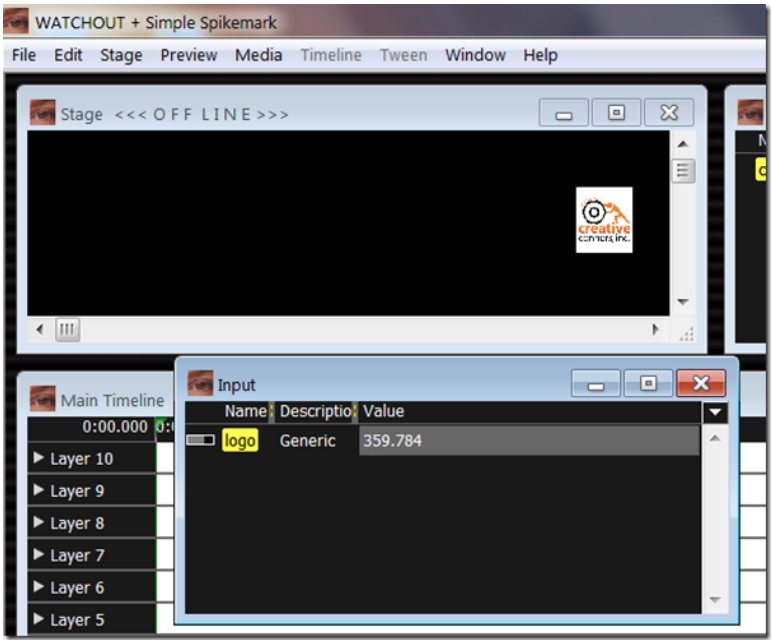


Figure 368

I hope this gives you a little inspiration to create some stunning stage effects. This tutorial is just a taste of what can be achieved when Spikemark and WATCHOUT are used together in live theatre. As you start using this feature in production, please let us know how it works for you and send us some video. We love to see this stuff in action.

Troubleshooting

When automation works well, it's a fun and wondrous technology. When automation doesn't work it's hair-pulling, frustrating black-magic. If you've made it to this part of the manual, I assume you are at the hair-pulling stage. I'll try to help answer some of the more common problems and solutions here. If these suggestions don't address your problem, give us a call, send an email, or use the discussion forums.

Contact Us

Phone 401-289-2942 (free tech support for the first 90 days, after that $30 / half-hour fees may apply at our discretion).

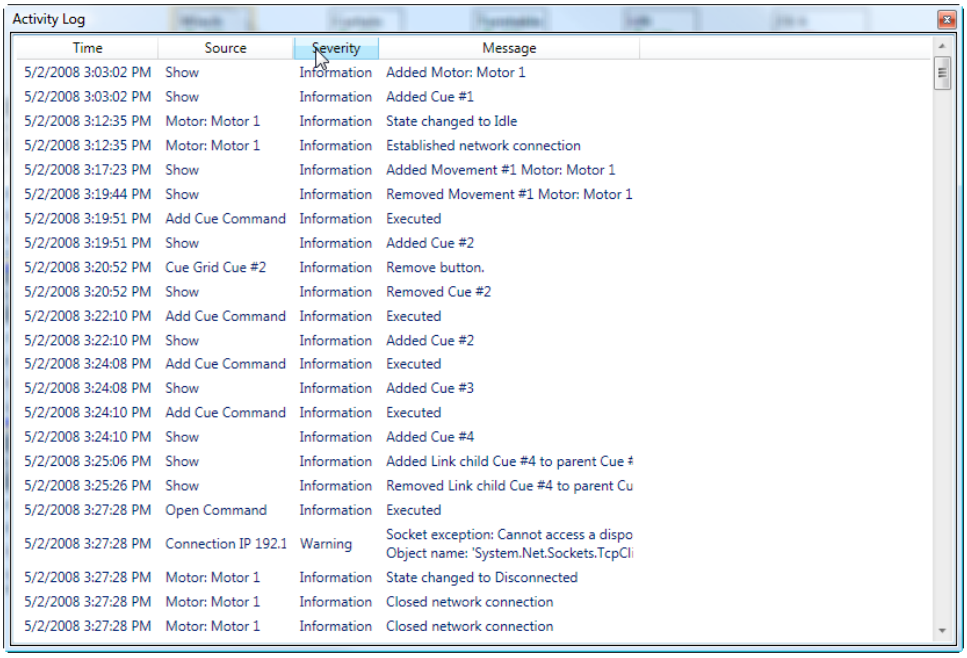
Email support@creativeconners.com (always free, response times vary from minutes to a couple of hours).

Access our Frequently Asked Questions web page at http://creativeconners.com/pages/faq.html

Web forums can be found at http://creativeconners.com/phpBB3/

Issue	Possible Solutions
The Stagehand Network Connection Doesn't Work	The most common reason for this is that the IP address between the Stagehand and the computer are not compatible. See the "Setting Up a Network Connection" section on page 41.
	Check your network cabling. Swap out the cables, preferably with a cable you know is good from another working Stagehand.
	Try resetting the IP address of the Stagehand to a slightly different address, and then update the address in Spikemark. If the network server in the Stagehand has stalled, resetting its IP address will get it started again.
	Make sure that you have a network hub between the computer and the Stagehand. If you are trying to connect directly from the computer to the Stagehand you need a special type of network cable—Crossover Cable.
	Swap out the network hub.

Issue	Possible Solutions

<table>
<tr><td rowspan="4">A Motor Didn’t Start To Move When I Ran A Cue</td><td>Check to see if the motor is sitting on a limit switch. Limit switch status is shown in the Cue Grid and the Stagehand Editing Pane.</td></tr>
<tr><td>Make sure you have a Proportional Gain of at least 1 in the Tuning section of the Stagehand Editing Pane.</td></tr>
<tr><td>Make sure that the cue had a speed and acceleration more than zero (0).</td></tr>
<tr><td>Check the activity log to get some information about what happened. In the Window menu, click Log Viewer. The Log Viewer is displayed and it shows all the actions that Spikemark is executing behind the scenes. You can sort the log by clicking on the column headers.
 
</td></tr>
<tr><td>Spikemark Showed A Strange Error Message Then I Had To Quit</td><td>That’s a bug, and we need to fix it. Please email a copy of your show file, a copy of the Activity Log, and a brief description of any pertinent information to support@creativeconners.com.</td></tr>
</table>

Issue	Possible Solutions
I Can't Find The Log File To Email It To Support	From the **Help** menu, click **Reveal Log File**. The **Application Data** folder is hidden by default on some Windows installations. To view the folder, click the **Start Menu** and select **Computer**. Press the **Alt** key to reveal the **File** menu. Click **Tools**, **Folder Options..., View** and then click **Show hidden files and folders**. Log files are stored in: C:\Users\spikemark\AppData\Roaming\Creative Conners, Inc\Spikemark\2.0.0.0
A Motor Starts To Run In A Cue But Stops Abruptly Before Getting To Spike	The motor isn't tuned well enough to reach its target position: **1.** Tune the motor better if possible. See the "Motor Tuning" section on page 79. **2.** In the **Cue Behavior** section of the **Stagehand Editing Pane,** deselect the **Strict Timing Mode** field. This will let the motor keep trying to get to target position even if the tuning is sub-optimal. **3.** If the motor is showing a **Position Error** fault and you are certain that nothing is physically stuck in the machinery, you can raise the **Max Position Error** value in the **Position** section of the **Stagehand Editing Pane**. **4.** If the motor is showing a **Position Error** fault, you can turn off the **Abort On Position Error** option inside the **Position** section of the **Stagehand Editing Pane**. This is not the ideal thing to do. If the encoder becomes broken or disconnected, nothing will stop the motor from running away. Since no feedback is being sent to the Stagehand, the Stagehand will send increasing amounts of power to the motor until it is at full speed and you will need to use the Emergency Stop to shutdown the motor. See the "Setting the Max Position Error" section on page 85 for more information. The motor is programmed to move at a speed beyond its physical ability: **1.** Run the motor on **Manual Control** at full speed and note the speed value displayed in **Speed** section of the **Stagehand Editor Pane**. **2.** Enter the maximum speed value witnessed in Step 1 into the **Max Speed** value in the **Stagehand Editor Pane**. **3.** Lower the cue speed.

The Motor Runs Full Speed In The Wrong Direction	The motor and encoder polarity are out of sync. Swap two of the power leads in the motor plug or swap the encoder signals A & /A with B & /B inside the encoder plug.
Spikemark Not Responding/ Running Slowly	Change the motor's refresh status rate. See the "Changing the Status Refresh Rate" section on page 90 for instructions.

Appendix A - Basic Motion Control Concepts

From the perspective of an audience member, motion control is magic. From the technician's perspective it should not be magic. The combination of Stagehand motor controllers and Spikemark software create a motion control system that everyone can use, but it is still important to know some basic concepts to avoid mistakes and help troubleshoot your system.

Power

Depending on the model of Stagehand that you are using, you need to supply either 120vac or 230vac power to each Stagehand. The Stagehand will convert this power into something suitable for your motor. The output voltage, or frequency for AC motors, is varied to achieve variable motor speed.

The Stagehand tracks the motor's position and then relays that position to Spikemark to display on-screen. If a Stagehand loses power, the next time it powers up it will reset its position to zero (0). This is very important, because if a motor is sitting 10' from centerline and the power is interrupted, when the power comes back on the Stagehand will think that the motor is now at 0' from centerline. In this case, the motor needs to be driven back to the true 0' spike (usually a piece of spike tape or limit switch) and reset to 0', either through Spikemark (using the **Reset Position** button in the **Position** section of the **Stagehand Editing Pane**) or by cycling power to the Stagehand.

The best operating procedure is to return all motors to zero (0) at the end of every performance. If power is lost overnight when no one is around, then it won't affect the Stagehand's position information. A pre-show check should be done every night to insure that every motor is truly at the correct position.

Since all position information is stored in the Stagehand, the PC running Spikemark can be shut down at any time without affecting position information whatsoever.

Position Feedback

How does the Stagehand know how to change speeds? Good question. There are two factors that affect the speed of a motor. The first is the target speed that is programmed for each cue. In Spikemark you tell each motor how fast to move in each cue. This information is sent to the appropriate Stagehand, which in turn supplies some voltage to the motor and watches to see what happens. If the motor moves faster than the Stagehand expected, it reduces the voltage. If the motor moves slower than expected, the voltage is raised. This cycle continues at a rapid pace until the motor reaches the desired position.

In order for the Stagehand to know how far and how fast the motor is moving in relation to the voltage it is supplying, it needs an encoder. An encoder is a simple device that creates electronic pulses as it spins. The encoder is physically connected to the motor so that as the motor spins the encoder spins. If the encoder were disconnected from the motor, the Stagehand would not see any movement regardless of whether the motor was actually rotating or not. This could cause the motor to race, because the Stagehand would keep supplying more and more voltage to the motor hoping to see an increase in speed. A similar situation would occur if the polarity of the motor was reversed from the encoder.

There are safety features that you can set up to prevent a motor from racing, such as **Max Position Error**, which tells a Stagehand to shutdown if it doesn't get good response from the motor. This feature is discussed in the "Setting the Max Position Error" section on page 85.

Another safety precaution to guard against motor racing is discussed in the "Over-Travel Limits" section below.

Motor Tuning

The cycle of watching encoder pulses and adjusting motor voltage is controlled by the tuning parameters of the Stagehand. These tuning parameters are often referred to as the PID filter, which stands for Proportional, Integral, and Derivative gains. You enter numeric values for this filter into Spikemark, which will then send the filter parameters to a Stagehand. While the mathematics of this filter is somewhat complex, the practical application is not. We walked through motor tuning in greater detail in the "Motor Tuning" section on page 79.

Over-Travel Limits

If the encoder position is wrong because of loss of power, physical disconnect from the motor, or some equipment failure, a physical limit switch can be used to protect the motor from running too far. These limit switches, when struck, will prevent the motor from moving any farther in the detected direction (either forward or reverse). The Stagehand will only permit a movement in the opposite direction until the limit switch is cleared.

These basic principles of motion control will become more concrete and intuitive as you become more familiar with the operation of Spikemark and Stagehand.

Index

3-Dimensional Stage View, 18
Abort On Position Error, 85
Acceleration, 125, 128, 129
Active Mode
  controlling with an FX, 90
  cue links and, 88
  for an FX, 111
  safety concerns for a motor and, 88
  safety concerns for an FX and, 110
  setting for a motor, 88
  setting for an FX, 109
Add Link Button, 135, 137, 138, 139
Add Movement Button, 164, 169
Add Spike Button, 165
Adding
  a new show, 174
  cues, 122
  FX movements to a cue, 130
  motor movement to a cue, 123
  motors, 65
  new FX, 103
Advanced
  section of Cue Grid, 90
Auto-repair, 117, 119
Child Cue, 134, 135, 137, 138, 139
  illustrated version of, 135
Completed Cue
  what happens after, 184
Completion Link, 136
  creating, 135
  illustrated version of, 135
  when not to use, 143, 144
Configuring
  FX, 105
  Motors, 65
Confirming Encoder Feedback, 51
Connecting
  to a motor, 118
  to a motor or FX, 49
  to an FX, 118
Creative Conners
  contact information, 207
  online tutorials website, 14
  technical support email address, 207
  web forums, 207
Cue Behavior
  for a motor, 88
  for an FX, 109
Cue Completion Mode
  strict timing, 86
  target tolerance, 86
Cue Controller, 57, 61, 181, 183, 184, 185
Cue Description, 122, 123
Cue Grid, 61, 66
  rearranging, 176
Cue Links, 134–44
  child cues, 134
  cue completion, 135
  deleting, 140
  infinite loops, 135, 143
  motor position, 138
  multi-speed/multi-position, 141
  parent cues, 134
  time link, 137
  using FX inputs, 139
Cue Number, 122, 123, 187, 188
Cue Outline Color Meaning
  blue, 185
  green, 183
  pink, 180
  red, 180
Cue Restore, 189
Cue Restoring, 189
Cues
  adding, 122
  adding a motor movement to, 123
  adding an FX movement to, 130
  changing how fast a motors moves within, 127
  changing motor position in, 126
  changing total time of, 129
  completed, 184
  deleting, 132
  linking, 134
  loading, 177
  pile on, 134

removing a motor's movement from, 131
restoring, 189
running, 56, 181
what happens while running, 182
writing, 55
Curtain Model
making a, 156
Deactivated
FX, 110, 188
motor, 89, 187
Deactivating FX Input Action, 112
Delay Field, 137
Delete Movement, 131
Derivative Gain, 80, 212
Disconnect From All, 120
Disconnecting
from a motor, 119
from an FX, 119
Distance from Center, 152, 155, 158, 159
Distance from Plaster, 152, 155, 158, 159, 160
Elevation From Stage, 152, 155, 158
Enable All Links, 190, 192
Encoder Counts
decreasing, 52
increasing, 52
increasing vs. decreasing, important notes, 70
Encoder Feedback
confirming, 51
Error Message
in SpikeMark, 208
Ethernet link
confirming that it works, 41
Filter Loaded, 82
Forward Limit Status, 187
Forward Physical Limit, 78
Frequently Asked Questions Website, 207
Front View Button, 148
FX
activating outputs, 109
active mode, 109, 111
adding a new, 103
adding movement to a cue, 130
configuring, 105
connecting to, 118
connecting to, quick start, 49
deactivating action for, 112
disconnecting from, 119
Input Action, 112
inputs, 131
outputs, 131
removing input action for, 115
renaming, 105
renaming inputs and outputs, 106
soft stopping action for, 113
status icon descriptions, 108
status information in cue grid, 188
FX Editing Pane, 103
FX Input Link
creating, 139
FX Status, 108, 188
GO Button, 181
Height, 155, 157, 159
Icons
for FX status, 108
for motor status, 68
Input Action
for an FX, 112
removing for an FX, 115
Input Statuses, 188
Inputs, 131
renaming for an FX, 106
Installing SpikeMark, 8
Microsoft's .Net 3.5 framework, 13
registering software, 13
Integral Gain, 81, 212
Integral Limit, 82
IP Address, 117
illustrated version of, 41
rules for setting, 41, 44
segments of, 41, 44
Setting for your computer, 41
unique for each stagehand and computer, 41
Isometric View Button, 149
Jump To Cue button, 178
Left View Button, 148
Length, 155, 157, 159
Lift Model
making a, 153
Limits

forward, 76
physical, 78
reverse, 76
rules for setting, 77
Linking Cues, 134–44
Load Cue Button, 56, 57
Load Movement Button, 190
Loaded Cue Indicators, 57
Loading a Cue, 177
Log File, 209
Lurching
what to do if motor is, 80
Main Screen, 61
Manual Controls, 51, 69, 73, 76
Manually Jogging a Motor on Screen, 69
Max Forward Position, 54, 76
rules for setting, 77
Max Position Error, 85
maximum allowable, 85
setting, 85
Max Speed, 129
Min Reverse Position, 54, 76, 152
rules for setting, 77
Motion Control Concepts, 211–12
Motor Tuning, 212
Over-Travel Limits, 212
Position Feedback, 211
Power, 211
Motor, 65–92
active mode, 88
adding a new, 65
adding movements to cues, 123
changing position of in a cue, 126
changing speed automatically, 142
changing speed manually, 141
configuring, 65
connecting to, 118
connecting to, quick start, 49
deactivated, 88, 187
disconnecting from, 119
doesn't start, 208
lurching, what to do if, 80
manually jogging with on screen controls, 69
movement parameters, 124
removing movement from a cue, 131
renaming, 67
resetting position value for, 78
runs full speed in wrong direction, 210
scaling position, 71
selection for a position link, 139
smooth out motion for, 81
speed in cue, 127
starts but stops abruptly, 209
tuning, 59, 84
what to do after motor is tuned, 85
Motor Cue Completion Mode, 86
Motor Editing Pane, 61, 66
using to add a spike, 167
Motor Name, 187
Motor Position, 187
changing using 'ping-pong' effect, 143
Motor Status, 187
deciphering during a show, 187
icon descriptions, 68
Motor Tuning, 79–84, 212
flow chart, 84
Movements
visual indicators of primed and ready, 179
what happens when complete, 184
Multi-position Cue Links, 141
Multi-speed Cue Links, 141
Network Connections
connecting, 116
disconnected, 116
errors, 116
Network Connections
setting up a, 41
Network Connections
auto-repair, 117
Network Connections
auto-repair, 119
Network Connections
breaking, 119
Network Connections, 116–20
Network Connections
status for a motor, 187
Network Connections
status for an FX, 188
Network Connections
not working, 207
Network Settings

editing, 116
Network Status, 50
New Show
creating, 174
Newest Features in SpikeMark
3-dimensional stage view, 18
spikes, 15
Stage Model Viewer, 18
streamlined interface, 16
time-based cues, 16
Next Cue Button, 178
Nightly Motor Position Resetting, 78, 211
Online Tutorials, 14
Outputs, 109, 131
manually activating for an FX, 109
renaming for an FX, 106
statuses of, 188
Over-Travel Limits, 212
Parent Cue, 134, 136, 137, 138, 139
illustrated version of, 135
Parts of Main Screen
Cue Controller, 61
Cue Grid, 61
Motor Editing Pane, 61
Stage Model Viewer, 61
Physical Connections
illustrated example of, 40
setting up, 40
Physical Limits, 78
when to set, 40
PID Filter, 212
Pile On Cues, 134
Ping Pong
a motor's position, 143
Plan View Button, 147
Polling Interval, 90, 91
Poorly Tuned
what to do when motor is, 84
Position, 124, 126
changing in a cue, 126
Position Feedback, 211
Position Link, 143, 144
creating, 138
illustrated example of, 135
Position Scale, 72, 73, 74
Position Units, 73
Power, 211
Power Loss
what to do, 78
Preset View Buttons, 147
Previous Cue Button, 178
Proportional Gain, 59, 80, 212
Quick Start Guide, 18–60
Ramp Time, 125, 126, 128, 130
Rearrange Stagehand Order, 176
Rearranging the Cue Grid, 176
Remove Cue, 132
Renaming
a Motor, 67
an FX, 105
inputs and outputs for an FX, 106
Reset Position, 79
Reset Position Button, 72, 74
Resetting Motor Position Each Night, 78
Restoring a Cue, 189
Reverse Limit Status, 187
Reverse Motor Polarity, 70
Reverse Physical Limit, 78
Right View Button, 149
Rotation X, 154
Rotation Y, 152, 161
Rotation Z, 154, 159
Running Cues, 56, 181
indicators of, 58
what happens while, 182
Running Shows, 173–92
deciphering motor status during, 187
restoring a cue when, 189
Safety, 7
Sampling, 81
Scaling Factor
setting, 71
Schematic, 150, 154, 156, 160
scrollbars
using in the Stage Model Viewer, 147
Setting
FX active mode, 109
Max Position Error, 85
Min Reverse and Max Forward positions, 76
motor cue completion mode, 86
motor's active mode, 88

soft limits, 53
stage dimensions, 149
Setting Up
IP address for your computer, 41
Network Connections, 41
Physical Connections, 40
SpikeMark workspace, 174
Smooth Out a Motion, 81
Soft Limits
setting, 53
Soft Stopping
using an FX, 113
Speed, 125, 127, 128, 129
allowing SpikeMark to update, 128
changing directly, 127
changing for a motor automatically, 142
changing for a motor manually, 141
changing for motor in a cue, 127
Spike Button, 126
Spike Position Button, 165, 170
using to add a spike, 163
SpikeMark
error message, 208
frequently asked questions website, 207
is not responding, 210
main screen, 49, 61
Newest Features, 14
online tutorials, 14
technical support email address, 207
web forums, 207
Spikes, 15, 121
adding using spike position button, 163
adding using the Motor Editing Pane, 167
changing position of, 171
Splitter Panes, 62, 174
Stage Dimensions
setting, 149
Stage Model Viewer, 18, 61, 145–62
as pre-tech tool, 145
floating, 176
making a curtain model in, 156
making a lift model in, 153
making a turntable model in, 159
making a winch model in, 150
navigation of, 146
opening in a separate window, 146
scrollbars, 147
setting the stage dimensions in, 149
Stagehand Loses Power
what to do, 78
Status
FX, 108
motor, 68
Status Bar, 180, 184, 186
Status Information
for a motor in the cue grid, 187
for an FX in the cue grid, 188
Status Refresh Rate, 90
changing the, 90
Streamlined Interface, 16
Strict Timing Mode, 86
System Requirements, 8
Target Tolerance Mode, 86
Time
changing for cues, 129
Time Link, 137, 143, 144
creating, 137
illustrated version of, 135
Time-Based Cues, 16
Total Time, 125, 126, 128, 129
Troubleshooting, 193–210
Tuned
what to do after motor is, 85
what to do when motor is poorly, 84
Tuning
basic for a motor, 59
motors, 79–84
Turntable Model
making a, 159
Vertical Flying Curtain, 159
Visual Indicators of Primed and Ready Movements, 179
Web Forums, 207
Website
for frequently asked questions, 207
for online tutorials, 14
Width, 155, 157, 160
Winch Model
making a, 150
Workspace
setup, 174
Writing Cues, 55, 121–33

Zoom Slider, 146